memento

Ne tenez plus votre livre d'une main tout en pianotant de l'autre sur le clavier de votre micro, le premier tome de "Clefs pour MSX" est un mémento qui s'ouvre à la bonne page et permet au débutant comme au spécialiste d'accéder rapidement au système de base des MSX : brochages et connecteurs, jeu d'instruction du Zilog Z80, adresses ROM et RAM, etc. Cet ouvrage fait la synthèse des informations techniques sur le standard MSX, et donne à l'utilisateur la possibilité de programmer en langage machine tout en ayant sous la main les différents paramètres des instructions BASIC, les codes ASCII, etc.

CLEFS POUR MSX Tome 1

9 782865 952434

Editions du PSI
B.P. 86
77402 Lagny/Marne
France

ISBN : 2.86595.243.6

Couverture Mariane de Nayer — Imprimé en France — 150 FF

CLEFS POUR MSX Tome 1

PSI

Rémy Pineau

memento

CLEFS POUR MSX

Tome 1

Système de base

Rémy Pineau

Editions du PSI

INTRODUCTION

Quel conducteur néophyte ou chevronné se séparerait du "livre de bord" de son véhicule, alors que celui-ci est indispensable pour tout bon entretien ?

Il en va de même en micro-informatique où il est essentiel de fournir à l'utilisateur des "outils" d'accès sûrs et efficaces à la "mécanique interne" d'un micro-ordinateur.

Le but principal des "clefs pour MSX" est de regrouper dans un seul document d'accès aisé, des informations techniques sur le standard MSX. Celles-ci sont généralement dispersées dans de nombreuses notices, manuels... et leur recherche en est longue et fastidieuse. Avec ce manuel, plus de problèmes de cet ordre. Ce mémento est (en quelque sorte) le "livre de bord" décrivant des rouages internes de tous les micro-ordinateurs répondant strictement aux critères de définition du standard MSX.

Il sera à ce titre particulièrement utile à tout possesseur d'un MSX qui envisage la programmation en langage machine Z80 où la connaissance détaillée des structures de la "MACHINE" est essentielle.

Le nouvel utilisateur en tirera également profit : un code ASCII oublié, un paramètre d'une instruction Basic mal interprété, et le mémento palliera la mémoire humaine défaillante.

Ce **premier tome** aborde la présentation du système, le langage Basic et les codes objet du Z80, les adresses ROM et RAM.

Le **deuxième tome** abordera la version étendue du MSX avec disques (MSX DOS, Basic disque et Utilitaires).

En route pour la première étape de ce long périple !

SOMMAIRE

SOMMAIRE

SCHEMA SYNOPTIQUE

Zone	Contenu	Adresse Hex	Adresse Dec	Page
ROM 32Ko — Bas de la mémoire	SYSTEME D'EXPLOITATION	0000H	0D	Page 0 SLOT 0
ROM 32Ko	INTERPRETEUR BASIC	4000H	16384D	Page 1 SLOT 0
RAM 32Ko	TABLE DES INSTRUCTIONS DE PROGRAMME (TIP)	8000H	32768D	Page 2 SLOT 0
RAM 32Ko	TABLE DES VARIABLES SIMPLES (TVS)			
RAM 32Ko	TABLE DES VARIABLES DIMENSIONNEES (TVT)			
RAM 32Ko	ZONE MEMOIRE LIBRE (ZML)			
RAM 32Ko	ZONE DE LA PILE OPERATIONNELLE (PO)	C000H	49152D	Page 3 SLOT 0
RAM 32Ko	ZONE DE STOCKAGE DES CHAINES DE CARACTERES (ZCC)			
RAM 32Ko	BLOC DE CONTROLE DE FICHIER (BCF)	F380H	62336D	
RAM 32Ko — Haut de la mémoire	ZONE DES VARIABLES SYSTEMES + VECTEURS (HOOK)	FFFFH	65535D	

Adresse Hex	Adresse Dec	E/S	Rôle	Périp.	Circuit
80	128	E/S	Ecriture/lecture des données sur 8251	RS232	8251
81	129	E	Lecture du status commande du 8251	RS232	8251
82	130	E	Lecture des interrupteurs vitesse	RS232	8251
83	131	E	Lecture des interrupteurs mode	RS232	8251
83	131	S	Ecriture du bit de masque d'interruption	RS232	
84	132	E/S	Compteur 0 du timer 8253	RS232	8253
85	133	E/S	Compteur 1 du timer 8253	RS232	8253
86	134	E/S	Compteur 2 du timer 8253	RS232	8253
87	135	S	Ecriture du mot de commande du 8253	RS232	8253
90	144	E	Lecture du signal BUSY de l'imprimante	IMPR.	
90	144	S	Ecriture du signal STROBE	IMPR.	
91	145	S	Ecriture du caractère sur l'imprimante	IMPR.	
98	152	E/S	Ecriture/lecture registre VDP (07)	VDP	TMS 9918A 9929A
99	153	E/S	Ecriture/lecture du registre d'état	VDP	
A0	160	S	Ecriture commande PSG	PSG	AY 8910-3
A1	161	S	Ecriture registre PSG	PSG	
A2	162	E	Lecture PORT 14 ou 15 PSG	PSG	
A8	168	E/S	Ecriture/lecture du PORT A du PPI	PPI	8255 utilisé en mode 0
A9	169	E/S	Ecriture/lecture du PORT B du PPI	PPI	
AA	170	E/S	Ecriture/lecture du PORT C du PPI	PPI	
AB	171	E/S	Ecriture/lecture du registre d'état	PPI	
B0 B8	176 184	E/S	Ecriture/lecture du crayon optique SANYO	crayon optique	
D0 D8	208 216	E/S	PORTS réservés au contrôleur de disque	disque	
F7	247	S	Ecriture du mot de commande couleur		

E = Entrée = Lecture
S = Sortie = Ecriture

MODES D'AFFICHAGE D'ECRAN (tableau récapitulatif)

Moniteur-TV

MODE		Résolution	Matrice nxm pixels	Nombre de config.	Nombre de couleurs	Lutins (Sprites)	Nombre de caractères
Graphique I * (Screen 1) +	MAX	256 x 192	8 x 8 (5)	256	16 (1)	oui	32 x 24
	NORMAL	240 x 192					29 x 24
Graphique II* (Screen 2) +	MAX	256 x 192	8 x 8	768	16 (2)	oui	32 x 24
	NORMAL	240 x 192					29 x 24
Multicolore * (Screen 3) +	MAX	256 x 192	4 x 4	-	16 (3)	oui	32 x 24
	NORMAL	240 x 192					29 x 24
Texte * (Screen 0) +	MAX	256 x 192	8 x 6 (4)	256	2 parmi 16	non	40 x 24
	NORMAL	240 x 192					39 x 24 37 x 24

→ NORMAL → 8 pixels espaces (blancs) horizontalement à droite et à gauche du caractère.

MAX → pas d'espace (capacité maximale d'affichage).

* Mode de fonctionnement du VDP.

\+ Mode d'affichage sous Basic.

(1) Deux couleurs maximum par groupe horizontal de 8 pixels.

(2) Une couleur d'avant-plan + une couleur de fond pour chaque caractère dans un groupe de 8 caractères consécutifs.

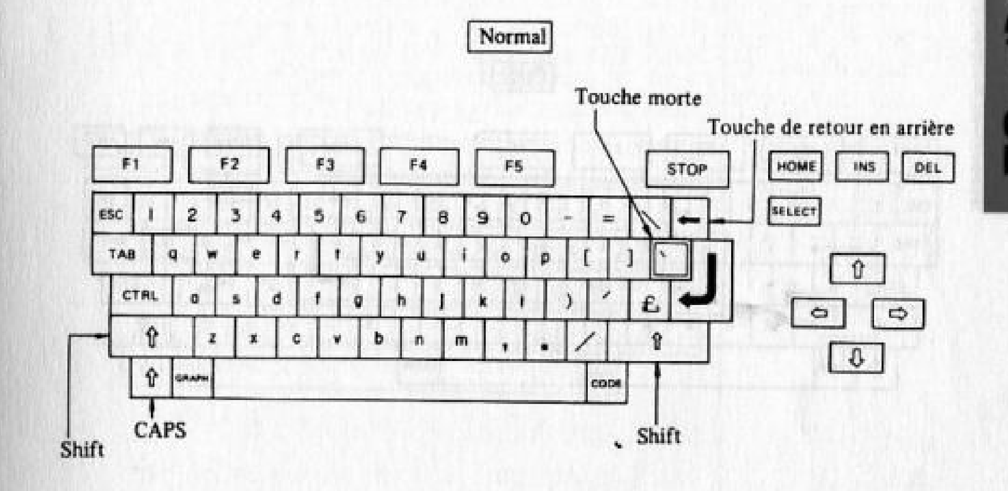
(3) Pas de restriction dans la proximité des couleurs.

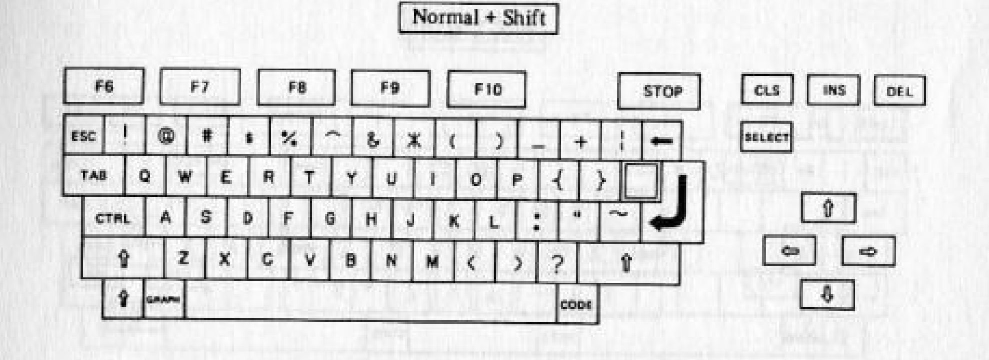
(4) Admet uniquement du caractère alphanumérique (ou semi-graphique) inscrit dans une matrice de 5 x 7 ; tronque les caractères semi-graphiques 8 x 8.

(5) Admet tous les caractères alphanumériques 5 x 7 ou semi-graphiques jusqu'à 8 x 8.

PRESENTATION

Rôle des touches
(exemple : Y15 503F YAMAHA)

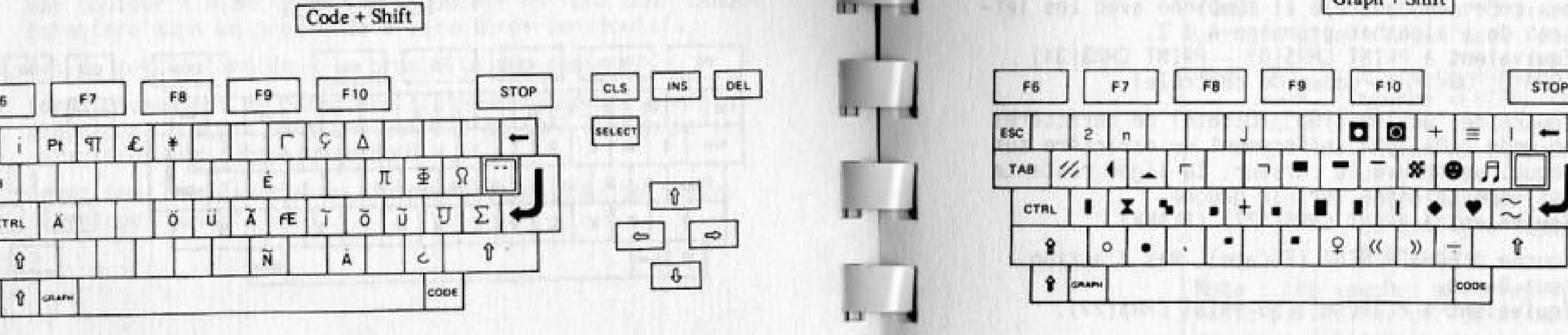




Rôle des touches (suite)

Code

Code + Shift



Rôle des touches (suite)

Graph

Graph + Shift

Rôle des touches spéciales

Touche	Rôle
<↵>	Touche <ENTER> ou <RETURN> ; prise en compte d'une ligne d'instructions Basic, provoque à l'affichage ou à l'impression un "Retour Chariot" et un "Interligne" (CR+LF). **Equivalent** à <CTRL><M> ou PRINT CHR$(13).
<↑> <←> <→> <↓>	Touches de déplacement de positionnement du curseur en mode "COMMANDE" ou "EDITION".
<BS>	Touche de retour en arrière (BACKSPACE). **Equivalent** à <CTRL><H> ou PRINT CHR$(8).
<CAPS>	Touche de verrouillage en mode "MAJUSCULES" (CAPitalS). **Equivalent** à <SHIFT> <lettre>(allumage, voyant lumineux).
<CLS>	Voir <HOME> ou <SHIFT><HOME>=<CLS>.
<CODE> <n>	Touche permettant l'accès au jeu supplémentaire 1 de caractères alphanumériques (lettres grecques, lettres accentuées si combinée avec la touche <n> (n : numérique ou alphanumérique).
<CTRL> <A> à <Z>	Touche de CONTROLE (ConTRoL) permettant l'accès aux codes de contrôle si combinée avec les lettres de l'alphabet primaire A à Z. **Equivalent** à PRINT CHR$(0) - PRINT CHR$(31) (0-31D, 00-1FH → codes de contrôle).
<DEL>	Touche de "DESTRUCTION" (DELete) de caractères en mode "EDITION" (effacement du caractère sur lequel se trouve le curseur, la ligne restante est repositionnée vers la gauche. **Equivalent** à PRINT CHR$(127) (BLANK).
<ESC>	Touche d'ECHAPPEMENT (ESCape). Pas d'action sous Basic. **Equivalent** à <CTRL><[> ou PRINT CHR$(27).
<HOME>	Touche de retour du curseur en haut à gauche de l'écran (en mode minuscule ou NORMAL). En mode <SHIFT><HOME> = <CLS>, le retour du curseur en haut à gauche de l'écran s'accompagne de l'effacement de tout caractère affiché (CLear Screen). **Equivalent** à <CTRL><K> ou PRINT CHR$(11) pour <HOME>. <CTRL><L> ou PRINT CHR$(12) pour <CLS>.

Touche	Rôle
<INS>	Touche d'INSERTION (INSert) de caractères en mode "EDITION". Les caractères entrés après la frappe de cette touche sont insérés à partir de la position courante du curseur (cette commande est annulée si une quelconque des touches de direction du curseur est utilisée). **Equivalent à** <CTRL><R> ou PRINT CHR$(18).
<F1> à <F10>	Touche de fonction prédéfinie. Les touches <F1> à <F5> sont utilisables en mode "NORMAL". Les touches <F6> à <F10> en mode <SHIFT>, c'est-à-dire : <Fn+5>=<SHIFT><Fn>. Les fonctions initiales de ces touches sont les suivantes : <F1> : COLOR[b] <F2> : AUTO[b] <F3> : GOTO[b] <F4> : LIST[b] <F5> : RUN<↵> <F6> : COLOR 15,4,7 ou COLOR 15,4,4 <F7> : CLOAD <F8> : CONT <↵> <F9> : LIST.<↵>[p] <F10> : <CLS> RUN <↵> COLOR 15,4,4 : couleur d'avant plan = blanc, arrière plan + bordure = bleu foncé ; COLOR 15,4,7 : couleur d'avant plan = blanc, arrière plan = bleu foncé, bordure = cyan. Avec [b] : blanc ou espace ; <↵> : touche <ENTER> (touche <RETURN>) <CLS> : touche effacement de l'écran (et retour du curseur en haut à gauche de l'écran) ; [p] : positionnement du curseur en début de la ligne Basic (n° d'instruction). **Nota** : ces touches sont reprogrammables sous Basic par KEY n,"cc" ou cc = chaîne de caractères pour définition fonction.
<GRAPH>	Touche de sélection des jeux de caractères semi-graphiques (jeu n° 1 →<GRAPH> <X>, jeu n° 2 → <SHIFT> <GRAPH> <X>.
<SELECT>	Touche de SELECTION (SELECT). Non utilisé sous Basic. **Equivalent** à <CTRL><X> ou PRINT CHR$(24).

Touche	*Rôle*
<SHIFT><HOME> =<CLS>	Touches de sélection du mode <CLS> (effacement écran + <HOME>).
<SHIFT><x>	Clavier normal (mode "MAJUSCULES").
<SHIFT><CODE> <x>	Jeu n° 2 optionnel de caractères alphanumériques.
<SHIFT><GRAPH> <x>	Jeu n° 2 de caractères semi graphiques.
<SHIFT><Fn>	Touche de sélection des fonctions préprogrammées F6 à 10.
<STOP>	Touche d'interruption temporaire d'exécution de programme (redémarrage par frappe de la même touche).
<CTRL><STOP>	Arrêt d'exécution de programme, retour en mode "COMMANDE", affichage du message "Break in nnnn" où nnnn est le numéro de ligne Basic où l'exécution a été interrompue (reprise d'exécution possible pour la commande CONT).
<TAB>	Touche de déplacement du curseur (TABulation horizontale) sur une même ligne, huit caractères vers la droite. **Equivalent** à <CTRL><I> ou PRINT CHR$(9).

Matrice de décodage

PORT B: PB0, PB1, PB2, PB3, PB4, PB5, PB6, PB7
(8255) PPI PORT C: PC0, PC1, PC2, PC3
A B C D — 0 1 2 3 4 5 6 7 8 9

	X7	X6	X5	X4	X3	X2	X1	X0
Y0	7	6	5	4	3	2	1	0
Y1	;	]	[	\	=	-	9	8
Y2	B	A	/	.	,	~	'	
Y3	J	I	H	G	F	E	D	C
Y4	R	Q	P	O	N	M	L	K
Y5	Z	Y	X	W	V	U	T	S
Y6	F3 F8	F2 F7	F1 F6	CODE	CAPS	GRAPH	CTRL	SHIFT
Y7	↵	SELECT	BS	STOP	TAB	ESC	F5 F10	F4 F9
Y8	⇨	⇩	⇧	⇦	DEL	INS	HOME	SPACE
Y9								

Description

Le clavier comporte une matrice de neuf lignes (Y0 à Y8) et de huit colonnes (X0 à X7) ; ceci permet donc de disposer de 9 x 8 = 72 touches.

Pour lire une matrice, on transmet sur le 1/2 octet inférieur du port C (C0-C3) le numéro de la ligne à analyser. On procède alors à la lecture du port B (B0-B7).

Si une touche est enfoncée, le bit Bn considéré passe à l'état 0 (les autres bits restant à 1).

Exemple

Touche <N> enfoncée → bit B3 passe à 0 ; ce bit B3 correspondant à la touche <N> pourra être lu lorsque la ligne Y4 sera écrite dans le port C.

CARACTERISTIQUES DES SIGNAUX DE DONNEES CASSETTE

Accès entrée	Connection au Jack "ECOUTEUR".
Accès sortie	Connection au Jack "MICROPHONE".
Méthode de synchronisation FORMAT	START-STOP (asynchrone) (1 bit START + 8 bits données + 3 bits STOP).
Rapidité de modulation (R)	**1200 baud** - valeur par défaut. Donnée "0" : 1200 Hz - 1 cycle Donnée "1" : 2400 Hz - 2 cycles 0 1 0 1200 Hz 1e forme d'onde — 2400 Hz 2e forme d'onde **2400 baud** - valeur choisie par logiciel (SCREEN ou CSAVE) Donnée "0" : 2400 Hz - 1 cycle Donnée "1" : 4800 Hz - 2 cycles 0 1 0 1 2400 Hz 1e forme d'onde — 4800 Hz 2e forme d'onde
Type de modulation	FSK (Frequency Shift Keying) type KANSAS CITY
Connecteur	DIN 45326 - 8 points.

SORTIE TV

Côté ordinateur
Prise + embase DIN 8 broches 45326

brochage spécifique à chaque constructeur *

Broche	Signal	Connexion
1	Masse	7 8 6 / 3 1 / 5 2 4
2	Son	
3	AV(+12V)	
4	Y	
5	YS(+4V)	
6	R	
7	V	
8	B	

* Pour Yamaha YIS 503F

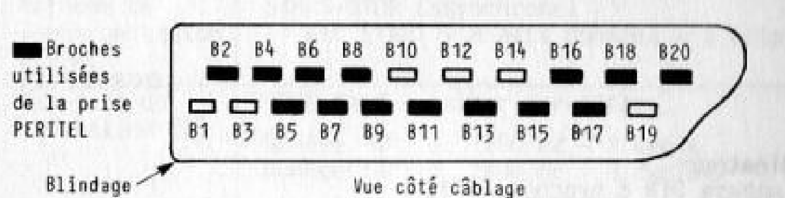
Broche	Signal
1	B
2	YS(+4V)
3	Masse
4	R
5	Son
6	V
7	AV(+12V)
8	Y

*Pour Sanyo PHC 28

Broche	Signal	Connexion
1	YS(+4V)	7 8 6 / 3 1 / 5 2 4
2	Masse	
3	B	
4	Y	
5	R	
6	AV(+12V)	
7	Son	
8	V	

*Pour Canon V20

Côté téléviseur
Prise + embase PERITEL 21 points



Masse

- B_1 Sortie "Audio", voie de droite
- B_2 Entrée "Audio", voie de droite → Son
- B_3 Sortie "Audio", voie de gauche
- B_4 Masse commune "Audio"
- B_5 Masse "Bleu"
- B_6 Entrée "Audio", mono ou voie de gauche → Son
- B_7 Entrée composante "Bleu" → B
- B_8 Entrée "Commutation lente" → AV (+12V) +
- B_9 Masse "Vert"
- B_{10} Horloge
- B_{11} Entrée composante "Vert" → V
- B_{12} Commande à distance
- B_{13} Masse "Rouge"
- B_{14} Masse "Commande à distance"
- B_{15} Entrée composante "Rouge" → R
- B_{16} Entrée "Commutation rapide" → YS (+4V) +
- B_{17} Masse "Vidéo"
- B_{18} Masse "Commutation rapide"
- B_{19} Sortie "Vidéo"
- B_{20} Entrée "Vidéo" (+ Synchro) → Y
- B_{21} Blindage de la prise

+ : ACTIF POUR ECRAN PERITELEVISION

Côté ordinateur
Prise + embase DIN 8 broches 45326

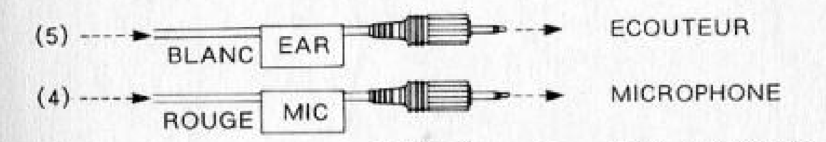
- brochage identique pour tous modèles

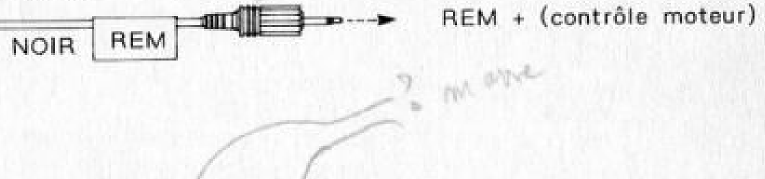
MICROPHONE →
ECOUTEUR →
CONTROLE MOTEUR →

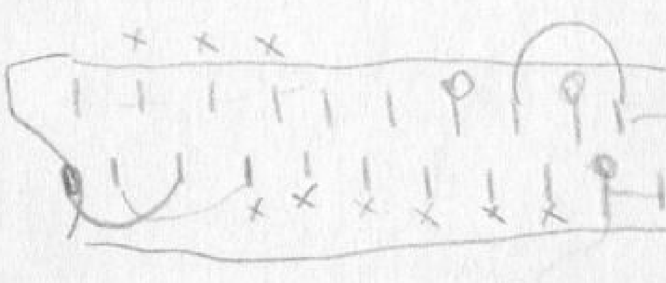


Broche	Signal	Connexion
1	Masse	
2	Masse	
3	Masse	
4	Sortie cassette	
5	Entrée cassette	
6	REM+	
7	REM-	
8	Masse	

Côté cassétophone
Jack : 3,5 mm
REM+: 2,5 mm







INTERFACE JOYSTICK
(embase AMP 9 broches)

Broche	Signal	Connexion
1	Avant	
2	Arrière	
3	Gauche	
4	Droite	
5	+5V	
6	Bouton 1	
7	Bouton 2	
8	Sortie	
9	Masse	

INTERFACE IMPRIMANTE
(embase AMPHENOL 14 broches)

Broche	Signal	Connexion
1	Validation données	(DATA STROBE)
2	PDB0	
3	PDB1	
4	PDB2	
5	PDB3	
6	PDB4	
7	PDB5	
8	PDB6	
9	PDB7	
10	NC	
11	Occupé	(BUSY)
12	NC	(Non Connecté)
13	NC	(" ")
14	Masse	

FENTE PAR CARTOUCHE ROM
(connecteur femelle 50 broches)

N°	Nom	I/O	N°	Nom	I/O	N°	Nom	I/O
1	CS1	O	2	CS2	O	3	CS12	O
4	SLTSL	O	5	N/C	-	6	RFSH	O
7	WAIT	I	8	INT	I	9	M1	O
10	BUSDIR	I	11	IORQ	O	12	MERQ	O
13	WR	O	14	RD	O	15	RESET	O
16	N/C *	-	17	A9	O	18	A15	O
19	A11	O	20	A10	O	21	A7	O
22	A6	O	23	A12	O	24	A8	O
25	A14	O	26	A13	O	27	A1	O
28	A0	O	29	A3	O	30	A2	O
31	A5	O	32	A4	O	33	D1	I/O
34	D0	I/O	35	D3	I/O	36	D2	I/O
37	D5	I/O	38	D4	I/O	39	D7	I/O
40	D6	I/O	41	GND	-	42	CLOCK	O
43	GND	-	44	SW1	-	45	+5V	-
46	SW2	-	47	+5V	-	48	+12V	-
49	SOUNDIN	I	50	-12V	-			

I → E = entrée

O → S = sortie

Broche		*Signification*
CS1	:	ROM 4000H-7FFFH - signal de sélection
CS2	:	ROM 8000H-FFFFH - signal de sélection
CS12	:	ROM 4000H-BFFFH - Signal de sélection
SLTSL	:	Signal de sélection de SLOT
RFSH	:	Signal de rafraîchissement
WAIT	:	Signal d'attente vers CPU
INT	:	Signal de demande d'interruption
M1	:	Signal du cycle "aller chercher" du CPU
BUSDIR	:	Contrôle de la direction du bus de données extérieur quand la cartouche est sélectionnée (niveau 0 si données transmises par cartouche)
IORQ	:	Signal de demande d'entrée/sortie
MERQ	:	Signal de demande d'accès mémoire
WD	:	Signal d'écriture
RD	:	Signal de lecture
RESET	:	Signal de RAZ
A0-A15	:	Bus d'adresse
D0-D7	:	Bus de données
CLOCK	:	Horloge à 3,579 MHZ
GND	:	Masse
SW1,SW2	:	Insertion/extraction (détection pour protection)
SOUNDIN	:	Entrée sonore (-5dBm)
+5V	:	alimentations
+12V	:	alimentations
-12V	:	alimentations

* non connecté

Z-80A (CPU (1))

A11	1	40	A10
A12	2	39	A9
A13	3	38	A8
A14	4	37	A7
A15	5	36	A6
ø	6	35	A5
CD4	7	34	A4
CD3	8	33	A3
CD5	9	32	A2
CD6	10	31	A1
VDD	11	30	A0
CD2	12	29	VSS
CD7	13	28	RFSH
CD0	14	27	M1
CD1	15	26	RESET
INT	16	25	BUSRQ
NMI	17	24	WAIT
HALT	18	23	BUSAK
MREQ	19	22	WR
IORQ	20	21	RD

TMS9929A/TMS9918A (VDP (2))

(EUROPE)			(USA)
RAS	1	40	XTAL1
CAS	2	39	XTAL2
AD7	3	38	R-Y/CLOCK
AD6	4	37	GROM CLK
AD5	5	36	Y/COMVID
AD4	6	35	B-Y/EXTVID
AD3	7	34	RESET/SYNC
AD2	8	33	VDO
AD1	9	32	RD0
AD0	10	31	RD1
R/W	11	30	RD2
VSS	12	29	RD3
MODE	13	28	RD4
CSW	14	27	RD5
CSR	15	26	RD6
INT	16	25	RD7
CD7	17	24	CD0
CD6	18	23	CD1
CD5	19	22	CD2
CD4	20	21	CD3

µPD8255A (PPI)

PA3	1	40	PA4
PA2	2	39	PA5
PA1	3	38	PA6
PA0	4	37	PA7
RD	5	36	WR
CS	6	35	RESET
GND	7	34	D0
A1	8	33	D1
A0	9	32	D2
PC7	10	31	D3
PC6	11	30	D4
PC5	12	29	D5
PC4	13	28	D6
PC0	14	27	D7
PC1	15	26	VDD
PC2	16	25	PB7
PC3	17	24	PB6
PB0	18	23	PB5
PB1	19	22	PB4
PB2	20	21	PB3

AY-3-8910 (PSG)

VSS	1	40	VDD
NC	2	39	TESTI
ANALOG CHANNEL B	3	38	ANALOG CHANNEL C
ANALOG CHANNEL A	4	37	DA0
NC	5	36	DA1
IOB7	6	35	DA2
IOB6	7	34	DA3
IOB5	8	33	DA4
IOB4	9	32	DA5
IOB3	10	31	DA6
IOB2	11	30	DA7
IOB1	12	29	BC1
IOB0	13	28	BC2
IOA7	14	27	BDIR
IOA6	15	26	SEL
IOA5	16	25	A8
IOA4	17	24	A9
IOA3	18	23	RESET
IOA2	19	22	CLOCK
IOA1	20	21	IOA0

(1) Voir chapitre MICROPROCESSEUR Z80 pour explication brochage.
(2) Voir chapitre PROCESSEURS SPECIALISES pour explication brochage.

PSG (AY-3-8910)

Broche n°	*Nom de broche*	*E/S*	*Fonction*
1	VSS		- Masse.
2	NC		- Non connecté.
3,4	ANALOG CHANNEL B, A	S	- Canal de sortie B, A.
5	NC		- Non connecté.
6~13	IOB7, 6, 5, 4, 3, 2, 1, 0	E/S	- Port de données 8 bits.
14~21	IOA7, 6, 5, 4, 3, 2, 1, 0	E/S	- Port de données 8 bits.
22	CLOCK	E	- Horloge de référence pour la fréquence, le bruit et l'enveloppe.
23	$\overline{\text{RESET}}$	E	- Entrée RAZ.
24	A9	E	- Fixé à 0.
25	A8	E	- Fixé à 1.
26	$\overline{\text{SEL}}$	E	- Sélection de la fréquence d'horloge.
27	BDIR	E	- Contrôle des opérations internes.
28,29	BC2, BC1	E	
30~37	DA7, 6, 5, 4, 3, 2, 1, 0	E/S	- Entrée/sortie de données.
38	ANALOG CHANNEL C	S	- Canal de sortie C.
39	TEST1		- Broche de test.
40	VDD		- +5V.

PPI (μPD8255A)

Broche n°	*Nom de broche*	*E/S*	*Fonction*
1~4	PA3, 2, 1, 0		- Port A.
5	$\overline{\text{RD}}$	E	- Entrée de lecture.
6	$\overline{\text{CS}}$	E	- Sélection boîtier.
7	GND		- Masse.
8,9	A1, A0	E	- Entrée signal pour sélection registre interne.
10~17	PC7, 6, 5, 4, 0, 1, 2, 3		- Port C.
18~25	PB0, 1, 2, 3, 4, 5, 6, 7		- Port B.
26	VDD		- +5V.
27~34	D7, 6, 5, 4, 3, 2, 1, 0	E/S	- Bus de données.
35	RESET	E	- Entrée RAZ.
36	$\overline{\text{WR}}$	E	- Entrée d'écriture.
37~40	PA7, 6, 5, 4		- Port A.

CARACTERISTIQUES GENERALES

Données

- *Entier* : -32768 à +32767
- *Simple précision* : $\pm$9,99999 E-64 à $\pm$9,99999 E+62.
- *Double précision* : $\pm$9,9999999999999 E-64 à $\pm$9,9999999999999 E+62.
- *Chaîne* : 0 à 255 caractères.
- *Numéro de ligne* : 0 à 65529 inclus.
- *Longueur d'une ligne de programme* : 255 caractères maximum.

Occupation de la mémoire

- *Ligne de programme* : minimum cinq octets. Deux octets pour le numéro de ligne, deux octets pour le pointeur vers la ligne suivante, un octet pour marquer la fin de ligne (00). De plus, chaque mot-clé occupe un ou deux octets, tous les autres caractères occupent un octet.
- *Une variable entière* : cinq octets (deux pour la valeur, deux pour le nom de la variable et un pour le type).
- *Une variable simple précision* : sept octets (quatre pour la valeur, deux pour le nom de la variable et un pour le type).
- *Une variable double précision* : onze octets (huit pour la valeur, deux pour le nom de la variable et un pour le type).
- *Une variable de chaîne* : six octets minimum (un pour le type, deux pour le nom, deux pour l'adresse et un pour la longueur) + un octet par caractère.
- *Une variable tableau* : douze octets minimum (deux pour le nom, un pour le type, deux pour la taille, un par nombre de dimension, deux pour chaque dimension et quatre, six, huit ou seize (selon le type de la variable) pour chaque élément du tableau).
- *Une boucle FOR NEXT* : vingt-cinq octets.
- *Un GOSUB actif* : cinq octets.
- *Un niveau de parenthèses* : seize octets (quatre octets plus douze octets pour stocker un résultat intermédiaire).

Caractère utilisé	Rôle	PRINT USING A$;N A$	N	Résultat
#	Etendu du champ numérique (position d'un chiffre)	### ### ###	13 2 -2	ƀ13 ƀƀ2 ƀ-2
.	Position du point décimal	##.## ##.### ##.###	1.2 1.2 -1.2	ƀ1.20 ƀ1.200 -1.200
+	Position du signe avant ou après le nombre (affiche en queue suivant signe N si +,mis en queue).	+#.# +#.# #.#+ #.#+	-1.123 1.123 -1.123 1.123	-1.1 +1.1 1.1- 1.1+
-	Position du signe après le nombre si N < 0	#.#- #.#-	-1.123 1.123	1.1- 1.1ƀ
**	Astérisques en tête	**#.## **#.##	23.53 2.53	*23.53 **2.53
$$	Signe dollar flottant avant le premier chiffre significatif	$$##.## $$##.## $$##.##	123.53 12.53 1.25	$123.53 ƀ$12.53 ƀƀ$1.25
**$	Combinaison de * et $$	**$##.## **$##.## **$##.##	123.53 12.35 1.23	*$123.53 **$12.35 ***$1.23
↑↑↑↑	Format exponentiel (notation scientifique)	#.##↑↑↑↑ ##.##↑↑↑↑	51235 51235	0.51E+05 ƀ5.12E+04
!	Premier caractère d'une chaîne	! !	"1234" "ABCD"	1 A
%%	2 premiers caractères d'une chaîne	%%	"ABCD"	AB
%..%	Chaîne de 2 caractères + le nombre d'espaces inclus	%ƀƀƀ%	"ABCDEFG"	ABCDE

ƀ = *blanc = espace*

Légende

b = paramètre optionnel (soulignement)
[ ] = paramètre optionnel
(1) = +9,99...10^{62}
(2) = -9,99...10^{62}
(3) = -32768
0 = opérateur
(4) = -65536
(5) = -149,668031044 61
(6) = 6,11...10^{13}
(1') = +9,99...10^{-64}
(2') = -9,99...10^{-64}
(3') = +32767
P/V = pseudo-variable
(4') = +65535
(5') = +145,0628608 5862
(d) = adresse de début RAM = 8000H = 32768D
(f) = adresse utilisable fin RAM = F37FH = 62335D

Mot-clé	Paramètres Valeur	Limites	Rôle	Type
ABS	(x)	(1')-(1) (2')-(2)	- Calcul de la valeur absolue de x.	F
AND	x,y	(3)-(3')	- "ET logique" entre x et y (x AND y = 1 si x=1 et u=1)	O/I
ASC	(cc)		- Détermination du code ASCII du 1er caractère de la chaîne cc.	F
ATN	(x)	(1')-(1)	- Calcul de l'arc tangente de x (x en radians).	F
AUTO	[d][,i]	0-65529	- Numérotation automatique des lignes d'un programme. d = n° de ligne de début de numérotation, i = incrément entre deux lignes consécutives (si AUTO → début = 10, incrément = 10).	C
BASE	n	0-19	- Indication de l'adresse de début des tables du VDP en VRAM suivant les quatre modes d'affichage (0≤m≤3) (5 tables par mode : noms, couleurs, générateur de configurations, attribut de lutin, configuration de lutins→0≤t≤4→n = m*5+t).	F
BEEP			- Emission d'un son court de longueur fixe et RAZ des registres du PSG et des tâches en attente de PLAY (équivalent à PRINT CHR$(7)).	I
BIN$	(x)	(3)-(4')	- Conversion d'un nombre entier x en la chaîne de caractères (CC) de son code binaire (maximum = 16 bits).	F
BLOAD	"C",R,X	(d)<X<(f)	- Chargement en RAM d'un module binaire de nom "CC" (code machine ou données) sauvegardé préalablement sur cassette par BSAVE (l'option R permet l'exécution automatique après chargement, X = adresse offset si présent permet l'implantation à l'adresse X en RAM).	C

BASIC

Mot-clé	Paramètres Valeur	Paramètres Limites	Rôle	Type
BSAVE	"CC",ad ,af,ae	CC=6 car. max.	- Sauvegarde sur cassette d'un fichier binaire en RAM (code machine ou donnée) de nom "CC". ad = adresse de début de la zone à sauvegarder. af = adresse de fin de la zone à sauvegarder. ae = adresse d'exécution de la zone à sauvegarder.	C
CALL	CC,(lp)	CC=15 car. max.	- Appel d'instructions extérieures étendant le Basic en ROM (0000-7FFF). CC = nom de la routine appelée. lp = liste de paramètres (optionnel).	I
CDBL	(x)	(1')-(1) (2')-(2)	- Conversion d'un nombre x en son équivalent double précision.	F
CHR$	(c)	0-255	- Retour du code de contrôle, du caractère alphanumérique, du caractère graphique dont le code ASCII est c.	F
CINT	(x)	(3)-(3')	- Retour du plus grand nombre entier inférieur ou égal à n (arrondi par défaut).	F
CIRCLE [STEP]	(x,y),r ,c,ad, ,af,a	(3)≤x≤(3') (3)≤y≤(3') (3)≤r≤(3') 0≤ c ≤15	- Tracé d'un cercle, d'un arc de cercle ou d'une ellipse. (x,y) = coordonnées du centre du cercle (absolues si STEP est omis, déplacements relatifs par rapport à la position du curseur si STEP est présent). r = rayon. c = couleur du tracé. ad = angle début du tracé (en radians). af = angle fin du tracé (en radians). a = rapport entre le rayon horizontal et le rayon vertical d'une ellipse.	I
CLEAR	n,h	n défaut = 200 octets	- Réservation de n octets de stockage de chaîne de caractères et initialisation de toutes les variables (à la mise sous tension automatiquement : CLEAR 200). Option h : haut de la mémoire Basic utilisable.	I

Mot-clé	Paramètres Valeur	Paramètres Limites	Rôle	Type
CLOAD	"CC"	6 car. max. pour CC	- Chargement d'un programme Basic à partir d'une cassette. 6 caractères peuvent être utilisés pour le nom CC.	C
CLOAD?			- Comparaison octet par octet du programme Basic sur cassette avec le programme résident en RAM (le programme en RAM est sauvegardé).	C
CLOSE	#n1,n2 ...nn	0-15	- Fermeture de tous les fichiers nommés par leur numéro : n1, n2... et libération des mémoires tampon associées (si aucun paramètre ne suit l'instruction, tous les fichiers sont fermés simultanément).	I
CLS			- Effacement de l'écran, déplacement du curseur en haut et à gauche (HOME).	
COLOR	a,f,b	0≤a≤15 0≤f≤15 0≤b≤15	- Sélection des couleurs utilisées pour l'affichage. a = couleur du texte ou du graphique (avant-plan). f = couleur du fond (ou arrière-plan). b = couleur de la bordure. Le choix porte sur une palette de 15 couleurs (des restrictions existent dans certains modes d'affichage : SCREEN 0, SCREEN 1, SCREEN 2). 0 = transparent 6 = rouge foncé 11 = jaune clair 1 = noir 7 = cyan 12 = vert foncé 2 = vert moyen 8 = rouge moyen 13 = magenta 3 = vert clair 9 = rouge clair 14 = gris 4 = bleu foncé 10 = jaune foncé 15 = blanc 5 = bleu clair ***Remarque*** : un paramètre au moins doit être présent ; le non emploi d'un paramètre est signalé par une <,>.	I

Mot-clé	Paramètres Valeur	Limites	Rôle	Type
CONT			- Redémarrage d'un programme interrompu par l'instruction STOP ou en pressant les touches <CONTROL><STOP>, <CONTROL><C>. CONT ne fonctionne pas si une correction a été effectuée après un arrêt.	C
COS	(x)	(1')-(1) (2')-(2)	- Calcul du cosinus de x (angle en radians).	F
CSAVE	"CC",v	6 car. max. pour CC	- Sauvegarde d'un programme Basic résident en RAM sur cassette. Le nom de fichier CC est assigné au programme, l'option v permet de choisir entre deux vitesses d'écriture (1 = 200 bauds, 2 = 2400 bauds). La vitesse d'écriture par défaut est 1200 bauds.	
CSNG	(x)	(1')-(1) (2')-(2)	- Conversion d'un nombre x en son équivalent simple précision.	
CSRLIN			- Indication de la position verticale du curseur (comprise entre 0 et 23). Utilisable seulement en mode TEXTE (SCREEN 0 ou SCREEN 1).	PV
DATA	x,y,z...		- Stockage de données dans une liste accessible par une instruction READ (les données peuvent être numériques ou caractères).	I
DEFDBL	x-y x,y,...		- Déclaration d'un ensemble de variables dans la plage x-y ou suivant la liste x,y... comme étant de type : DOUBLE PRECISION.	I
DEFFN	a(x1,x2)		- Définition d'une fonction créée par l'utilisateur. a = nom de la fonction x1,x2 = liste de variables ou paramètres	I
DEFINT	x-y x,y,...		- Déclaration d'un ensemble de variables dans la plage x-y ou suivant la liste x,y... comme étant de type : ENTIER.	I
DEFSNG	x-y x,y,...		- Déclaration d'un ensemble de variables dans la plage x-y ou suivant la liste x,y... comme étant de type : SIMPLE PRECISION	I

Mot-clé	Paramètres Valeur	Limites	Rôle	Type
DEFSTR	x-y x,y,...		- Déclaration d'un ensemble de variables dans la plage x-y ou suivant la liste x,y... comme étant de type : CHAINE DE CARACTERES.	I
DEFUSR	a	0-9	- Définition de l'adresse d'une routine en langage machine accessible à l'utilisateur. Dix routines sont accessibles (0≤a≤9) lorsque a n'est pas spécifié ; sa valeur par défaut est 0.	I
DELETE	nnnn-mmmm	0-65529	- Destruction de toutes les lignes d'un programme comprises entre les numéros inclus nnnn et mmmm (nnnn<mmmm) (si DELETE - nnnn destruction depuis le début jusqu'à nnnn).	C
DIM	(x,y, z...)		- Dimensionnement d'un tableau de 1 à n dimensions (dimensionnement automatique pour nombre d'éléments par dimension ≤ 11).	I
DRAW	"CC"	255 car.	- Instruction du MLG (Macro Langage Graphique) permettant la spécification rapide d'une série complexe de commandes graphiques à exécuter dans les modes d'affichage SCREEN 2 et SCREEN 3. La définition des commandes se fait à l'intérieur de la chaîne de caractères CC par l'intermédiaire de lettres majuscules abrégées de la commande en clair. *Commandes de déplacement* D(ownward)n →déplacement vers le bas (n positions). En →déplacement en diagonale vers le haut à droite (n positions). Fn →déplacement en diagonale vers le bas à droite (n positions). Gn →déplacement en diagonale vers le bas à gauche (n positions). Hn →déplacement en diagonale vers le haut à gauche (n positions).	I

Mot-clé	Paramètres Valeur	Limites	Rôle	Type
			L(eft)n →déplacement vers la gauche (n positions).	
			R(ight)n →déplacement vers la droite (n positions).	
			U(pward)n →déplacement vers le haut (n positions).	
			Commandes de position	
			B →déplacement du curseur sans traçage de points.	
			M x,y →déplacement du curseur au point de coordonnées x,y (x≤255 y≤191).	
			M +x,+y →déplacement relatif du curseur à partir de la dernière position absolue référencée x,y.	
			N →après déplacement du curseur, le curseur retourne à la dernière position référencée.	
			Commandes générales	
			A a →détermination de l'angle a sous lequel le dessin est affiché.	
			(0<a<3) →a=0 mode par défaut : le dessin est affiché normalement. a=1 le dessin subit une rotation de 90° dans le sens des aiguilles d'une montre. a=2 rotation de 180°. a=3 rotation de 270°.	
			C c →détermination de la couleur du tracé (0≤c≤15) la valeur par défaut est la couleur de l'avant-plan défini par COLOR.	
		1 < S < 255	S s →Détermination du facteur d'échelle (SCALE) pour l'effet de grossissement ou ZOOM (pour obtenir le facteur de grossissement il faut diviser S par 4). Par exemple, S=4→facteur d'échelle 4/4→1 échelle normale (valeur par défaut).	

Mot-clé	Paramètres Valeur	Limites	Rôle	Type
			X a$; →exécution de la sous-chaîne a$ à l'intérieur de la chaîne principale CC (le point virgule en fin d'expression est obligatoire).	
END			- Indication de la fin de l'exécution d'un programme et fermeture de tous les fichiers.	I
EOF	(n)	0-15	- Indication de la rencontre d'une fin de fichier (End Of File) pour un numéro de fichier n ouvert (n doit être inférieur à MAXFILES).	F
EQV	x,y	(3)-(3')	- Opérateur logique d'"<u>EQUIVALENCE</u>" entre deux opérateurs x et y (x EQV y = 1 si x = 1 et y = 1 ou si x = 0 et y = 0)	0/I
ERASE	a,<u>b</u>,<u>c</u>		- Elimination d'un ou plusieurs tableaux a,b,c de la mémoire RAM (après cette instruction, un tableau peut être redimensionné sans provoquer une erreur de type REDIMENSIONNED ARRAY.	I
ERL			- Retour du numéro de ligne dans laquelle une erreur est intervenue.	F
ERR			- Si une erreur est intervenue, ERR retourne une valeur liée au code d'erreur c.	F
ERROR	(c)	0-255	- Simulation de l'erreur spécifiée par le code c (1≤c≤23).	I
EXP	(x)	(5)-(5')	- Calcul de l'exponentielle de x (antilogarithme à base e de x).	F
FIX	(x)	(1)-(1') (2)-(2')	- Troncature de tous les digits d'un nombre à droite du point décimal (retour de la partie entière avec suppression de la partie fractionnaire).	F
FN	a		- Appel de la fonction a préalablement définie par DEF FN (s'utilise alors comme n'importe quelle autre fonction standard COS ATN).	F
FOR... **TO-STEP**	n1 to n2 STEP n3	(1')-(1) (2')-(2)	- Ouverture d'une boucle de programme (incrément de la boucle spécifiée par la valeur donnée à STEP).	I

Mot-clé	Paramètres Valeur	Limites	Rôle	Type
FRE	(x) ("cc") (c$)	(1')-(1) (2')-(2)	- Détermination du nombre d'octets de mémoire libre (x valeur numérique quelconque). - Détermination de l'espace cc disponible en nombre d'octets cc = chaîne de caractères ou variable c$ quelconque)	F
GOSUB	nnnn	0-65529	- Transfert du contrôle de programme au sous-programme débutant au numéro de ligne nnnn.	I
GOTO	nnnn	0-65529	- Transfert du contrôle de programme au numéro de ligne nnnn.	I
HEX$	(x)	(3)-(4')	- Conversion d'un nombre entier x en la chaîne de caractères (cc) représentative de son code hexadécimal (maximum = 4 digits hexadécimaux).	F
IF...THEN ...ELSE	(IF)expression1 (THEN)expression2 (ELSE)expression3		- Instruction de test conditionnel (si expression 1 est vérifiée. ALORS exécution de l'expression, 2, AUTREMENT exécution de l'expression 3).	I
IMP	x,y	(3)-(3')	- Opérateur logique d'IMPLICATION entre deux opérandes x et y (x IMP y = 1 si x=1 et y=0).	0/I
INKEY$			- Prise en compte d'un caractère clavier (si clavier utilisé lors de l'exécution d'un programme).	F
INP	(P)	0-255	- Entrée d'une valeur à partir d'un port spécifié.	F
INPUT	"a$"; l,m,n		- Entrée de données au clavier, suivant la liste l,m,n (E, SP, DP, CC) (le message a$ est optionnel).	I
INPUT$	(n)		- Lecture d'un nombre n de caractères entrés au clavier (sans affichage à l'écran).	I
INPUT$	(n1,#n2)		- Lecture d'un nombre n1 de caractères saisis sur le fichier associé au buffer n2.	I
INPUT#	n,l,m,p		- Saisie de données suivant la liste l,m,p sur le fichier de numéro spécifié n, ces données sont assignées à des variables et saisies dans le buffer n.	I

Mot-clé	Paramètres Valeur	Limites	Rôle	Type
INSTR	(n,a$,b$)	0 ≤n≤ 255	- Recherche de la sous-chaîne b$ à l'intérieur de la chaîne a$ à partir de la position n et affichage de la position du premier caractère de b$ dans a$ (0 est retourné si b$ n'est pas trouvé).	F
INT	(x)	(1')-(1) (2')-(2)	- Détermination du nombre entier le plus grand, inférieur ou égal à x (arrondi par défaut).	
INTERVAL OFF(ON)			- Desactivation (Activation) du déroutement déclaré par l'instruction ON INTERVAL GOSUB.	I
INTERVAL STOP			- Mémorisation du déroutement déclaré par l'instruction ON INTERVAL GOSUB mais inhibition de son exécution jusqu'à la rencontre de INTERVAL ON.	I
KEY	n,"cc"	cc=15 car. max.	- Assignation de la chaîne de caractères cc à la touche de fonction programmable dont le numéro n est spécifié (1≤n≤10). cc peut être une constante, une variable ou une expression.	C
KEY LIST			- Listage à l'écran (par ordre de numéro croissant) des chaînes de caractères associées aux dix touches de fonctions programmables.	C
KEY OFF			- Suppression de l'affichage sur la 24e ligne de l'écran de l'affichage des cc associées aux touches de fonctions programmables.	I
KEY ON			- Rétablissement de l'affichage des touches de fonctions programmables supprimé par KEY OFF, à la mise sous tension, la valeur par défaut est KEY ON).	I
KEY OFF	(n)	1-10	- Désactivation du déroutement déclaré par l'instruction ON KEY GOSUB.	I
KEY ON	(n)	1-10	- Activation du déroutement déclaré par l'instruction ON KEY GOSUB.	I

Mot-clé	Paramètres Valeur	Paramètres Limites	Rôle	Type
KEY STOP	(n)	1-10	- Mémorisation du déroutement déclaré par l'instruction ON KEY GOSUB, mais inhibition de son exécution jusqu'à la rencontre de KEY ON.	I
LEFT$	(cc,n) (A$,n)	0≤n≤255	- Détermination d'une sous-chaîne de caractères constituée des n caractères les plus à gauche de la cc ou de la variable caractère A$.	F
LEN	(cc) (A$)		- Détermination de la longueur (nombre de caractères constitutifs) de la chaîne de caractères cc ou de la variable A$.	F
LET			- Assignation d'une valeur à une variable (optionnel).	I
LINE	[STEP] (x1,y1) [STEP] (x2,y2) [,c] [,BF]	(3)≤x1,x2≤(3') (3)≤y1,y2≤(3') 0≤c≤15	- Tracé d'un segment, rectangle, rectangle coloré entre les points de coordonnées (x1,y1) et (x2,y2). Si STEP est omis, le segment est tracé entre les points de coordonnées absolues. Dans le cas contraire, (x1,y1) et (x2,y2) définissent un déplacement relatif sur l'échelle des coordonnées par rapport à la position courante du curseur. c = couleur du dessin. Si l'option B (box) est utilisée, un rectangle dont la diagonale est fixée par (x1,y1) et (x2,y2) est tracé. Si BF (box filled) est utilisé, le rectangle sera coloré dans la couleur c ou suivant la couleur en vigueur fournie par COLOR.	I
LINE INPUT#	["a$"] ;cc		- Entrée d'une chaîne de caractères cc au clavier (quel que soit son contenu). Le message a$ affiché à l'écran est optionnel.	I
LINE INPUT#	n,cc	cc=255 car. max.	- Lecture des caractères du fichier ouvert n (0≤n≤15) associé à un buffer de même numéro (utilisation en E/S cassette).	I
LIST	nnnn- mmmm	0-65529	- Liste des lignes de programme comprises entre nnnn et mmmm incluses (écran).	C

Mot-clé	Paramètres Valeur	Paramètres Limites	Rôle	Type
	-mmmm	0-65529	- Liste des lignes de programme depuis le début jusqu'à la ligne mmmm incluse (écran).	
	nnnn-	0-65529	- Liste des lignes de programme depuis la ligne nnnn incluse jusqu'à la fin du programme (écran).	
LLIST	nnnn- mmmm	0-65529	- Impression des lignes de programme comprises entre nnnn et mmmm.	C
	-mmmm	0-65529	- Idem LIST mais sur imprimante.	
	nnnn-	0-65529	- Idem LIST mais sur imprimante.	
LOAD	"cc"[,R]		- Chargement d'un programme Basic préalablement sauvegardé sur le périphérique spécifié dans cc (CAS : par le cassétophone). L'option R permet l'exécution automatique du programme après chargement. cc en complément du nom de périphérique utilisé indique également le nom du programme à charger.	C
LOCATE	x,y,b	0≤x≤255 0≤y≤255	- Positionnement du curseur à un endroit quelconque de l'écran en mode texte (SCREEN 0, SCREEN 1) suivant les coordonnées x,y. b=0, le curseur est éteint et n'apparaît pas à l'écran ; b=1, le curseur est allumé.	I
LOG	(x)	0<x≤(1)	- Calcul du logarithme à base e de x (logarithme naturel).	F
LPOS	(x)	(1')≤x≤(1) (2')≤x≤(2)	- Indication de la position de la tête d'impression du buffer d'imprimante (x est un argument fictif pouvant prendre n'importe quelle valeur dans les limites fixées).	F
LPRINT	x,y,z		- Impression d'une donnée ou d'une liste de données (x,y,z) sur imprimante.	I
LPRINT TAB	(x)	0-255	- Impression avec positionnement du curseur sur n'importe lequel des 40 points possibles de la ligne déterminée par x à partir du bord gauche de l'écran (si x>39), le curseur saute x\40 lignes et se positionne à la colonne X MOD 40.	I/C

Mot-clé	Paramètres Valeur	Paramètres Limites	Rôle	Type
LPRINT USING	expression		- Impression formattée de nombres et de chaînes de caractères, sur imprimante *(voir tableau spécifique pour la liste des différents types de champs utilisés)*.	I
MAXFILES	=n	0-15	- Spécification du nombre n de fichiers pouvant être ouverts simultanément en mémoire (réservation du nombre de mémoires tampons associées au contrôle de ces fichiers).	PV
MERGE	"cc"		- Fusionnement d'un programme résidant en mémoire avec un programme sur cassette (format ASCII) cc = expression alphanumérique contenant le nom du périphérique et du programme sur ce périphérique.	C
MID$	(cc,pos ,long)	0≤pos≤255 0≤long≤255	- Détermination d'une sous-chaîne de caractères à partir de la chaîne principale cc, la position de début de cette sous-chaîne est définie par le paramètre pos, la longueur par le paramètre long (si ce paramètre n'est pas spécifié, toute la sous-chaîne à partir de pos est retournée).	F
MOD			- Opérateur MODULO, calcul du reste d'une division entière (3 MOD 2 = 1).	O/I
MOTOR OFF			- Arrêt du moteur du cassétophone (fermeture du relais de commande à distance du cassétophone).	I
MOTOR ON			- Démarrage du moteur du cassétophone (ouverture du relais de commande à distance du cassétophone).	I
NEW			- Destruction du programme courant en mémoire et initialisation de toutes les variables.	C
NEXT			- Instruction de délimitation d'une boucle FOR.	I
NOT	x	(3)-(3')	- Opérateur logique de <u>NEGATION</u>. sur un opérande x (NOT $\overline{x}$ = x)	O/I
OCT$	(x)	(3)-(4')	- Conversion d'un nombre entier x en la chaîne de caractères (cc) de son code octal (max = 6 digits en octal).	F

Mot-clé	Paramètres Valeur	Paramètres Limites	Rôle	Type
ON... GOSUB	expression nnnn,mmmm,...	0-65529	- Branchement calculé multiple vers des sous-programmes déterminés par les numéros de lignes nnnn,mmmm, etc., en fonction de la valeur de : expression.	I
ON... GOTO	expression nnnn,mmmm	0-65529	- Branchement calculé multiple vers des lignes de programme repérées par les numéros nnnn,mmmm, etc., en fonction de la valeur de : expression.	I
ON ERROR GOTO	nnnn	0-65529	- Mise en place d'un sous-programme de piégeage d'erreur (si nnnn = numéro de ligne ≠ 0, si nnnn = 0, dévalidation de la routine) (déroutement vers le numéro de ligne nnnn si une erreur est rencontrée dans le programme).	I
ON INTER-VAL GOSUB	t nnnn	1≤t≤(4') 0-65529	- Branchement toutes les t unités de temps à un sous-programme débutant au numéro de ligne nnnn (unité de temps = 1/50 seconde). Si le déroutement est effectué un INTERVAL STOP est exécuté, ce branchement est validé par INTERVAL ON.	I
ON KEY GOSUB	mmmm ,nnnn ,oooo ,....	0-65529	- Branchement multiple vers des numéros de lignes mmmm,nnnn, oooo,... suivant la touche de fonction programmable utilisée (F1a F10) (1er numéro ligne → F1, 2e numéro de ligne → F2, etc.). Ce branchement est validé par KEY(n) ON.	I
ON SPRITE GOSUB	nnnn	0-65529	- Branchement vers un sous-programme débutant en nnnn, si une collision de deux lutins a été détectée. Ce déroutement est validé par SPRITE ON.	I
ON STOP GOSUB	nnnn	0-65529	- Branchement vers un sous-programme débutant en nnnn à chaque fois que les touches <CONTROL><STOP> ont été utilisées pour interrompre un programme. Ce déroutement est validé par STOP ON.	I

Mot-clé	Paramètres Valeur	Paramètres Limites	Rôle	Type
ON STRIG GOSUB	mmmm, nnnn, oooo, pppp, qqqq,	0-65529	- Déroutement de l'action sur le bouton poussoir d'une manette de jeux (JOYSTICK) se traduisant par le branchement vers des sous-programmes débutant en mmmm, nnnn, oooo, pppp, qqqq (1er numéro ligne → touche barre d'espace du clavier) (2e numéro ligne → bouton de la manette n° 1) (3e numéro ligne → bouton de la manette n° 2) (4e numéro ligne → bouton de la manette n° 1) (5e numéro ligne → bouton de la manette n° 2). Un numéro de ligne peut être omis (pas d'interception), son absence étant signalée par une virgule.	I
OPEN ...FOR ...A$#	"cc" m n		- Ouverture d'un fichier séquentiel vers le périphérique spécifié (périphérique devant être un des dispositifs logiques acceptés par Basic : CAS:, CRT:, LPT:, GRP). cc → précise le périphérique et le nom optionnel du fichier. m → précise le mode de fonctionnement du fichier. m=OUTPUT → le fichier est ouvert en écriture. m=INPUT → le fichier est ouvert en lecture. m=APPEND → le fichier est ouvert en écriture, les données entrées étant ajoutées à la fin du fichier. n → caractérise le numéro de fichier (ou de buffer associé) pour toutes les instructions d'E/S (valeur max est définie par MAXFILES).	I
OR	x,y	(3)-(3')	- Opérateur "OU Logique" entre les opérandes x et y (x OR y = 1 si x=1, y=1 OU x=1, y=0 OU x=0, y=1).	0/I
OUT	p,v	0-255	- Transmission de l'octet de valeur v (0≤v≤255) vers le port d'E/S p. Cette instruction permet le contrôle éventuel des circuits PPI, PSG, VDP.	I

Mot-clé	Paramètres Valeur	Paramètres Limites	Rôle	Type
PAD	(n)	0-7	- Lecture de l'état d'un accessoire de type manette de jeux (JOYSTICK). n<4 (pour manette de jeu n° 1). 4≤n≤7 (pour manette de jeu n° 2). n=0 ou n=4 → la fonction retourne -1 si une des huit directions de la poignée est utilisée. 0 dans le cas contraire. n=1 ou n=5 → la fonction retourne la valeur de la coordonnée horizontale (x) où la manette de jeu déplace le curseur. n=2 ou n=6 → la fonction retourne la valeur de la coordonnée verticale (y) où la manette de jeu déplace le curseur. n=3 ou n=/ → retourne l'état du bouton poussoir (-1 si bouton enfoncé, 0 si bouton non enfoncé).	F
PAINT [STEP]	(x,y) [,c1] [,c2]	0≤x≤255 0≤y≤191 0≤c1,c2≤15	- Coloriage d'une figure fermée quelconque. Ce coloriage s'effectue dans la couleur définie par c1 ($0<c_1<15$) jusqu'à ce que le "pinceau" rencontre un point de la même couleur, le point de départ du coloriage est défini par les coordonnées (x,y). Si c1 n'est pas mentionné, la couleur du dessin est celle préalablement définie par COLOR. En mode d'affichage, SCREEN 2 (haute résolution graphique), le coloriage n'est toléré qu'avec un pourtour du dessin ayant la même couleur que celle du coloriage (pas de paramètre c2). En mode d'affichage SCREEN 3 (basse résolution graphique), le coloriage par la couleur c1 est autorisé même avec un pourtour de couleur différente. Mentionner dans ce cas c2 pour indiquer la couleur des limites du dessin. Si STEP est utilisé, le début du coloriage est relatif au dernier point graphique tracé.	I

Mot-clé	Paramètres Valeur	Paramètres Limites	Rôle	Type
PDL	(n)	1-12	- Lecture des états des ports "manettes de jeux". Si n est impair → manette n° 1 utilisée. Si n est pair → manette n° 1 utilisée. (si aucune manette n'est reliée, cette fonction retourne 255 ; en cas contraire, la valeur retournée est comprise entre 0 et 254).	F
PEEK	(n)	(3)-(4')	- Lecture de la valeur contenue à l'adresse mémoire n.	F
PLAY	cc1,cc2, cc3		- Commande du Macro Langage Musical (MLM) permettant la création de morceaux de musique sur trois voies indépendantes avec une étendue maximale de huit octaves. Chaque voie est programmée en utilisant une, deux ou trois (cc1, cc2, cc3) expressions alphanumériques composées de caractères appartenant au MLM et dont la liste est la suivante : A → note LA B → note SI C → note DO D → note RE E → note MI F → note FA G → note SOL (Ces notes sont données dans une octave spécifiée par O. Les demi-tons sont indiqués par : # ou + dièses, - bémols) Ln → longueur ou durée de référence de la note (1≤n≤64) définie par 1/n. L2 → pause, ronde L3 → blanche pointée L4 → demi-pause, blanche L6 → noire pointée L8 → soupir, noire L16 → demi-soupir, croche L32 → quart de soupir, double croche L64 → huitième de soupir, triple croche	I

Mot-clé	Paramètres Valeur	Paramètres Limites	Rôle	Type
			Mn → sélection de la fréquence de l'enveloppe (1≤n≤65535) du son (valeur par défaut n=255). On → sélection de l'octave en cours pour toutes les notes émises après la commande (1≤n≤8). Valeur par défaut n=4. Rn → émission d'un silence de longueur n (1≤n≤64). Valeur de n par défaut n=4. Sn → choix de la forme d'onde (SHAPE) pour l'enveloppe du son (0≤n≤15). Huit formes différentes sont possibles. Valeur par défaut n=1. Tn → fixation du TEMPO de la mélodie (nombre de noires émises à la minute)(32≤n≤255). Valeur par défaut n=120. Vn → réglage du volume de sortie (0≤n≤15) : n=0 pas de son émis n=15 maximum sonore n=8 valeur par défaut Xa$ → exécution de la sous chaîne a$ à l'intérieur d'une des trois chaînes principales cc1, cc2, cc3 (l'insertion de plusieurs sous-chaînes permet la création de mélodies de longueur > 255 caractères).	
PLAY	n	0-3	- Vérification de l'exécution d'une mélodie sous instruction PLAY. n = numéro du canal sonore (1≤n≤3). Si 1≤n≤3, la fonction renvoie -1 si le canal concerné est en exécution, 0 si ce n'est pas le cas. Si n=0, la fonction retourne -1 si n'importe quel canal est en exécution.	F

Mot-clé	Paramètres Valeur	Paramètres Limites	Rôle	Type
POINT	(x,y)	0≤x≤255 0≤y≤191	- Détermination de la couleur du point graphique de coordonnées (x,y) (0≤c≤15).	F
POKE	n,v	(3)≤n≤(4') 0≤v≤255	- Ecriture à l'adresse mémoire n de la valeur v (0≤v≤255).	I
POS	(x)	(1')-(1) (2')-(2)	- Indication de la position horizontale courante (0-39) du curseur (x est un argument fictif) en affichage mode texte (SCREEN 0 et SCREEN 1).	F
PRESET [STEP]	(x,y)[,c]	(3)≤x≤(3') (3)≤y≤(3')	- Extinction d'un point de coordonnées (x,y) dans la couleur du graphique du fond (background) définie par COLOR dans le cas où le paramètre c n'est pas utilisé. - Allumage d'un point de coordonnées (x,y) dans la couleur définie par le paramètre couleur c (0≤c≤15) (instruction identique à PSET dans ce cas). Si l'option STEP est utilisée, le point s'éteint ou s'affiche par un déplacement relatif par rapport à la position courante du curseur graphique. ***Remarque*** *:* utilisation de PRESET et PSET uniquement en mode d'affichage graphique : SCREEN 2, SCREEN 3	I
PRINT	x,y,z		- Affichage à l'écran d'une donnée ou d'une liste de données x,y,z (peut être remplacé en entrée clavier par ?). Les séparateurs des données sont : ; affichage d'une donnée tout de suite après la précédente. , affichage après tabulation de 14 caractères.	I
PRINT TAB	(x)	0-255	- Affichage avec positionnement du curseur sur n'importe lequel des 40 points possibles de la ligne, déterminé par x à partir du bord gauche de l'écran (si x > 39, le curseur saute x\40 lignes et se positionne à la colonne x MOD 40).	I

Mot-clé	Paramètres Valeur	Paramètres Limites	Rôle	Type
PRINT USING	expres-sion		- Affichage formatté de nombre et de chaînes de caractères (voir tableau spécifique pour la liste des différents types de champs utilisés).	I
PRINT#	n,l,m,p	0≤n≤15	- Ecriture d'une liste de données l,m,p dans un fichier séquentiel de numéro n (le fichier doit être en mode OUTPUT ou APPEND ouvert par l'instruction OPEN). Le fichier est défini sur le périphérique CRT:, LPT:, CAS : ou GRP :	I
PRINT# USING	n,l,m,p	0≤n≤15	- Idem à l'instruction précédente, mais possibilité d'écriture de chaînes de caractères complètes (sans limitations de ponctuations).	I
PSET [STEP]	(x,y)[,c]	(3)≤x≤(3') (3)≤y≤(3') 0≤c≤15	- Allumage d'un point graphique de coordonnées (x,y) dans la couleur déterminée par c (ou si c non mentionné dans la couleur définie par COLOR). Si l'option STEP est utilisée, le point s'allume pour un déplacement relatif par rapport à la position courante du curseur graphique.	I
PUT SPRITE [STEP]	p, (x,y) [,c][,n]	0≤p≤31 (3)≤x≤(3') (3)≤y≤(3') 0≤c≤15 0≤n≤63 0≤n≤255	- Affichage d'un lutin n à l'écran au point de coordonnées (x,y). Ce lutin doit avoir été préalablement défini à l'aide de SPRITE$(n). p : représente le numéro de plan où s'effectue l'affichage du lutin (0≤p≤31). Si STEP est omis, le coin supérieur gauche du lutin est situé au point de coordonnées absolues (x,y) ; dans le cas contraire, x et y indiquent un déplacement relatif par rapport à la position courante du curseur graphique. c : sert à définir la couleur du lutin (0≤c≤15). Si c est omis, la couleur choisie est celle de l'avant-plan sélectionné par COLOR.	I

Mot-clé	Paramètres Valeur	Paramètres Limites	Rôle	Type
			n : définit le numéro choisi pour le lutin par l'instruction SPRITE(n), suivant la taille du lutin défini par SCREEN. Ce numéro n peut varier de 0 à 255 (lutin 8x8) ou de 0 à 63 (lutin 16x16) ; si n est omis, la valeur de p est prise par défaut.	
READ	x,y,z		- Lecture des valeurs x,y,z contenues dans une instruction DATA.	I
REM			- Instruction "REMARK" (commentaire) demandant à l'ordinateur d'ignorer le reste de la ligne (abréviation=').	I
RENUM	[n][,a] [,i]	0-65529	- Renumérotation des lignes d'un programme. La renumérotation débute à l'ancien numéro : a qui prend alors la valeur du nouveau numéro : n. Deux numéros consécutifs sont séparés par l'incrément i. Si l'ancien numéro a est omis, la renumérotation démarre au début du programme. Si n est omis, la renumérotation démarre à la ligne 10. Si tous les paramètres sont omis, la renumérotation démarre à la première ligne du programme (a). Cette ligne prend le nouveau numéro 10 (n), l'incrément par défaut est 10.	C
RESTORE			- Transfert du pointeur de données à la position de la première donnée dans une instruction DATA.	I
RESUME	nnnn	0-65529	- Fin d'un sous-programme de piégeage d'erreur, nnnn représente le numéro de ligne ou l'exécution normale doit reprendre.	I
RETURN			- Retour de sous-programme et exécution de l'instruction suivant la ligne d'instruction GOSUB.	I
RETURN	nnnn	0-65529	- Retour de sous-programme et exécution de l'instruction nnnn.	I

Mot-clé	Paramètres Valeur	Paramètres Limites	Rôle	Type
RIGHT$	(cc,n)	0-255	- Détermination d'une sous-chaîne de caractères à partir de la chaîne principale cc, cette sous-chaîne est déterminée par les n caractères les plus à droite de la chaîne principale.	F
RND	(x)	(1')-(1) (2')-(2)	- Génération d'un nombre pseudo-aléatoire valeur comprise entre 0 et 1 si x > 0, du nombre précédent si x=0.	F
RUN			- Exécution d'un programme à partir de son début.	C
RUN	nnnn	0-65529	- Exécution d'un programme à partir de la ligne nnnn.	C
SAVE	"cc"	6 car. max pour cc	- Sauvegarde un programme en mode ASCII sur le périphérique spécifié dans cc (CSAVE sauvegardant sous forme compressée ou tokénisée). cc a le format suivant : nom de périphérique (CAS:, LPT:, GRP:, CRT:) nom de fichier (6 caractères max). nom de fichier peut être omis.	C
SCREEN	[,m][,l] [,e][,v] [,i]	0≤m≤3 0≤l≤3 0≤e≤255 1≤v≤2 0≤i≤255	- Définition des modes d'affichage d'écran et de différentes options d'entrée/sortie (tous les paramètres sont optionnels, ils peuvent être omis mais une virgule doit en signaler l'absence). m → mode d'écran (l'écran est vidé et tous les registres du VDP sont redéfinis suivant l'instruction BASE). m=0 → mode texte 1 sans bordure d'écran, 40x24 caractères, lutins interdits. m=1 → mode texte 2 (par défaut). Bordure d'écran acceptée (voir instruction COLOR), 32x24 caractères, lutins acceptés.	I

Mot-clé	Paramètres Valeur	Limites	Rôle	Type
			m=2 → mode graphique 1. Définition graphique de 256x192 pixels, lutins acceptés, chaque groupe de huit pixels peut avoir sa propre définition de couleur (avant et arrière-plan). Le texte est accepté après ouverture d'un fichier par GRP: m=3 → mode graphique 2 (multicouleur). Définition graphique de 64x48 pixels, lutins acceptés, pas de limitation de couleurs. l → taille des lutins l=0 → les lutins ont 8x8 pixels (256 def possibles). l=1 → agrandissement par 2 dans les deux directions par rapport à l=0. l=2 → les lutins ont 16x16 pixels (64 def possibles). l=3 → agrandissement par 2, par rapport à l=2. e écho sonore de touche e=0 → aucun son n'est produit à la frappe d'une touche. e≠0 → émission d'un son à la frappe d'une touche (valeur par défaut e≠0). v vitesse de déroulement de la bande magnétique de la cassette v=1 → vitesse de transfert la plus faible correspondant à 1200 baud (valeur par défaut). v=2 → vitesse de transfert la plus élevée correspondant à 2400 baud. i type d'imprimante i=0 → imprimante compatible avec MSX (impression de tous les caractères). i≠0 → imprimante non compatible avec MSX (valeur par défaut). Les caractères graphiques sont convertis en blanc.	

Mot-clé	Paramètres Valeur	Limites	Rôle	Type
SGN	(x)	(1')-(1) (2')-(2)	- Calcul du signe de x (-1 si x < 0, 0 si x=0, +1 si x > 0).	F
SIN	(x)	(1')-(1) (2')-(2)	- Calcul du sinus de x (x en radians).	F
SOUND	r,v	0≤r≤13 0≤v≤255	- Instruction permettant le contrôle direct du Générateur Sonore Programmable (PSG = Programmable Sound Generator). AY-3-8910 par accès aux 14 registres de ce circuit LSI (paramètre r = numéro du registre 0≤r≤13) en écriture (paramètre v indiquant la valeur à inscrire dans le registre r 0≤v≤255).	I
SPACE$	(l)	0-255	- Création d'une chaîne d'espaces blancs de longueur l spécifiée (identique à STRING$ (1, " ").	F
SPC	(l)	0-255	- Impression d'une chaîne de l blancs (s'emploie uniquement avec PRINT).	F
SPRITE OFF			- Désactivation de la routine de déroutement ON SPRITE GOSUB en cas de collision de lutins.	I
SPRITE ON			- Activation de la routine de déroutement ON SPRITE GOSUB en cas de collision de lutins (doit précéder l'instruction de déroutement).	I
SPRITE STOP			- Désactivation provisoire de la procédure d'interception ; si une autre collision intervient, elle est mémorisée mais non exécutée immédiatement, l'activation se faisant ultérieurement par SPRITE ON.	I
SPRITE$	(n)	0≤n≤255 ou 0≤n≤63	- Définition et examen des configurations de lutins. n = numéro de lutin (0≤n≤255 en SCREEN 1, 0≤n≤63 en SCREEN 2 ou 3).	P/V

Mot-clé	Paramètres Valeur	Limites	Rôle	Type
			Cette configuration de lutins de 8 octets ou 32 octets se fait dans une expression alphanumérique où les bits "1" ou "0" sont précisés. Le nombre et la taille des lutins étant définis par ailleurs par le paramètre l de l'instruction SCREEN. Si SPRITE$ (n) est utilisé à droite du signe = dans une expression, la pseudo-variable retourne les 8 ou 32 octets de VRAM spécifiés par n.	
SQR	(x)	(1')-(1) (2')-(2)	- Calcul de la racine carré de x.	F
..STEP..	n		- Indication de l'incrément (ou pas n) dans une instruction de boucle FOR...STEP...NEXT.	I
STICK	(n)	0-2	- Indication de l'état d'une manette de jeu, retourne une valeur comprise entre 0 et 8, suivant la direction où la manette est poussée. STICK(0) correspond aux touches fléchées du clavier. STICK(1) et STICK(2) correspondent respectivement aux manettes de jeu n° 1 et n° 2. Les différentes valeurs sont les suivantes : [1 8 2 7 0 3 6 5 4] (0 signifie que la manette de jeu n'est pas utilisée). L'utilisation combinée de STICK et STRIG permet la lecture complète de l'état des périphériques type "manette de jeu".	F
STOP			- Interruption de l'exécution d'un programme émission à l'écran du message "BREAK IN nnn" où nnnn est le numéro de ligne où STOP a été écrit, exécution d'une instruction BEEP (la frappe de <CONTROL><STOP> a le même effet).	I

Mot-clé	Paramètres Valeur	Limites	Rôle	Type
STOP OFF			- Désactivation de l'interception de la frappe des touches <CONTROL><STOP> (branchement ON STOP GOSUB interdit).	I
STOP ON			- Activation de l'interception de la frappe des touches <CONTROL><STOP> (branchement ON STOP GOSUB autorisé).	I
STOP STOP			- Suspension provisoire de l'interception de la frappe des touches <CONTROL><STOP>. La frappe de <CONTROL><STOP> est mémorisée, mais l'exécution du déroutement n'intervient qu'après la rencontre d'un nouveau STOP ON.	I
STRIG	(n)	0-4	- Lecture de la valeur du bouton poussoir (ou gâchette) des manettes de jeu. n indique le numéro de la manette : n=0 → clavier (barre d'espacement). n=1 → manette de jeu n° 1. n=2 → manette de jeu n° 2. La fonction retourne 0 si le bouton poussoir n'est pas enfoncé, -1 s'il l'est.	F
STRIG OFF			- Désactivation de l'interception de l'appui du bouton poussoir des manettes de jeu (branchement ON STRIG GOSUB interdit).	I
STRIG ON			- Activation de l'interception de la frappe du bouton poussoir des manettes de jeu (branchement ON STRIG GOSUB autorisé).	I
STRIG STOP			- Suspension provisoire de l'interception de l'appui du bouton poussoir des manettes de jeu. Cet appui est mémorisé, mais l'exécution du déroutement n'est effectuée qu'après la rencontre d'un nouveau STRIG ON.	I
STRING$	(l,c)	0<l<255	- Retour d'une chaîne de caractères de longueur l, constituée des caractères c, où c peut être le code ASCII ou un caractère de chaîne (écriture "c").	F

Mot-clé	Paramètres Valeur	Limites	Rôle	Type
STR$	(x)	(1')-(1) (2')-(2)	- Conversion d'une expression numérique x en une chaîne de caractères (fonction inverse de VAL(x)).	F
SWAP	x1,x2	(1')-(1) (2')-(2)	- Echange des valeurs de deux variables numériques ou alphanumériques en une seule opération (sans faire appel à une 3e variable).	I
TAB	(n)	0-255	- Tabulation (alignement des colonnes) dans une instruction PRINT (voir PRINT TAB). TAB(0) correspond à la 1ère colonne ou colonne 0. Cette fonction informe PRINT de la colonne n où doit être positionné le curseur.	F
TAN	(x)	(1')-(6) (2')-(6)	- Calcul de la tangente de x (x en radians).	F
..THEN..			- Instruction d'un saut conditionnel de type IF...THEN, ou IF...THEN-ELSE (voir IF...THEN-ELSE).	I
TIME			- Indication du temps de l'horloge interne ou réglage de celle-ci. La pseudo-variable TIME, placée à droite d'un signe = retourne une valeur courante sous forme d'un entier non signé compris entre 0 et 65535 (état du compteur interne 16 bits incrémenté tous les 1/50e de seconde). Si TIME est placé à gauche du signe égal, le compteur interne est réajusté à la valeur indiquée (valeurs comprises entre -32768 et 65535, les valeurs comprises entre -32768 et -1 étant converties au préalable en entiers non signés).	P/V
..TO..			- Indication de la valeur limite dans une instruction de boucle FOR...TO...NEXT (voir FOR).	I
TROFF			- Dévalidation du traçage des numéros de ligne (mode TRACE OFF).	I

Mot-clé	Paramètres Valeur	Limites	Rôle	Type
TRON			- Validation du traçage des numéros de ligne (mode TRACE ON). Dans ce mode, tous les numéros de ligne rencontrés sont affichés à l'écran sous le format nnnn.	I
..USING..			- Instruction d'impression ou d'affichage formattée (voir PRINT USING ou LPRINT USING).	I
USR	n(x)	0≤n≤9	- Appel d'un sous-programme "utilisateur" en langage machine. Un DEFUSR n doit avoir été préalablement exécuté pour définir l'adresse du point d'entrée de la routine "UTILISATEUR" (USER en anglais). 10 appels de fonction sont possibles. x = expression de type quelconque, mais dont la présence est obligatoire.	F
VAL	(cc)		- Evaluation de la valeur numérique correspondant à une chaîne de caractères cc (fonction inverse de STR$(x)).	F
VARPTR	(x)		- Calcul des pointeurs de variable x (adresses où le contenu de la variable x est stocké en mémoire RAM).	F
VARPTR#	n		- Calcul des pointeurs du "BLOC DE CONTROLE DE FICHIER" (BCF) du fichier #n. ***Remarque :*** * Dans le cas d'une **variable numérique** : VARPTR(x) donne une adresse qui se trouve trois octets après **le descripteur de variable** (début zone stockage de la valeur). * Dans le cas d'une **variable alphanumérique** : VARPTR(x) donne l'adresse du 1er octet descripteur de la variable cc (longueur de la chaîne). * Dans le cas d'un **fichier** : VARPTR #n donne le 1er octet du BCF (la mémoire tampon des données débutant neuf octets plus loin). - POUR LES DETAILS D'UTILISATION DE LA FONCTION VARPTR(x) VOIR LE CHAPITRE "ADRESSES RAM".	F

Mot-clé	Paramètres Valeur	Paramètres Limites	Rôle	Type
VDP	(n)	0-7	- Accès direct aux registres internes du VDP 9918/9929. Cette pseudo-variable permet la lecture et l'écriture des huit registres (7 registres d'écriture + 1 registre d'état) du processeur d'affichage vidéo. VDP(n) permet de faire directement ce que BASE(n) n'autorise qu'indirectement, c'est-à-dire : définition des différentes tables de la RAM ; définition des modes d'affichage ; lecture du registre d'état du VDP (interception de SPRITE, détection d'un 5e lutin).	P/V
VPEEK	(n)	0-16383	- Lecture d'un octet de données contenu à l'adresse n de la RAM VIDEO ou VRAM (adressée par le circuit VDP 9918/9928/9929). C'est la seule méthode de lecture directe de la zone 16K VRAM complètement déconnectée de la zone RAM utilisateur (elle-même accessible par PEEK(x)).	F
VPOKE	n,v	0≤n≤16383 0≤v≤255	- Ecriture d'un octet de données de valeur v à l'adresse mémoire n de la zone VRAM (0≤v≤16383). Cette commande s'effectue moins rapidement que son équivalent POKE (pour la zone RAM utilisateur) car la zone VRAM n'est pas directement accessible au CPU Z80, mais par les ports d'E/S.	I
WAIT	p,a,b	(3)≤p≤(4') 0≤a≤255 0≤b≤255	- Attente jusqu'à ce qu'un port d'E/S p atteigne une certaine valeur. Les données sur le port spécifié sont comparées par un "OU EXCLUSIF" avec a, puis le résultat avec une autre valeur b avec un "ET". Le processus est répété jusqu'à ce que la valeur produite par les comparaisons donne un résultat non nul (seul <CONTROL><STOP> peut arrêter l'exécution de WAIT).	I
WIDTH	(l)	1-40 ou 1-32	- Ajustage de la largeur d'écran en mode texte (SCREEN 0 ou SCREEN 1) (valeur par défaut : 37 ou 29).	I
XOR	x,y	(3)-(3')	- "OU EXCLUSIF" entre les opérandes x et y (x XOR y=1 si x=1, y=0 ou si x=0 y=1).	0/I

Codes de contrôle (00H-1FH) + 7FH
(0D-31D) + 127D

Code Hex	Code Dec	Touches associées	Equivalent	Standard	Rôle
00	0			NUL	Non utilisé
01	1	<CTRL><A>		SOH	En tête de caractère semi-graphique 0<n<31D (voir codes semi-graphiques).
02	2	<CTRL><B>		STX	Déplacement du curseur au début du mot précédent.
03	3	<CTRL><C>		ETX	Interruption d'exécution lorsque MSX Basic est en attente d'une entrée (INPUT, INKEY$).
04	4	<CTRL><D>		EOT	Néant.
05	5	<CTRL><E>		ENQ	Effacement de la fin de la ligne à partir de la position du curseur.
06	6	<CTRL><F>		ACK	Déplacement du curseur au début du mot suivant.
07	7	<CTRL><G>	BEEP	BEL	Emission d'un son court.
08	8	<CTRL><H>		BS	Déplacement d'une position vers l'arrière. Efface le caractère rencontré.
09	9	<CTRL><I>	<TAB>	HT	Tabulation horizontale (8 positions).
0A	10	<CTRL><J>		LF	Déplacement vers la ligne suivante (interligne = LF).
0B	11	<CTRL><K>	<HOME>	VT	Déplacement du curseur vers le coin supérieur gauche (HOME).
0C	12	<CTRL><L>	<CLS>	FF	Effacement de l'écran (et retour du curseur vers le coin supérieur gauche).
0D	13	<CTRL><M>	<RETURN> ou <↵>	CR	Déplacement du curseur ou début de la ligne suivante après passage en fin de ligne (Retour Chariot = RC).
0E	14	<CTRL><N>		SO	Addition de texte sur la ligne courante (le curseur est automatiquement placé en fin de ligne).

Code Hex	*Code Dec*	*Touches associées*	*Equiva-lent*	*Stan-dard*	*Rôle*
0F	15	<CTRL><O>		SI	Non utilisé.
10	16	<CTRL><P>		DLE	Non utilisé.
11	17	<CTRL><Q>		DC1	Non utilisé.
12	18	<CTRL><R>	<INS>	DC2	Mode "Insertion" de l'éditeur.
13	19	<CTRL><S>		DC3	Non utilisé.
14	20	<CTRL><T>		DC4	Non utilisé.
15	21	<CTRL><U>		NAK	Effacement total de la ligne où se trouve le curseur.
16	22	<CTRL><V>		SYN	Non utilisé.
17	23	<CTRL><W>		ETB	Non utilisé.
18	24	<CTRL><X>	<SELECT>	CAN	Equivalent à <SELECT> . Pas d'action en mode EDITION.
19	25	<CTRL><Y>		EM	Non utilisé.
1A	26	<CTRL><Z>		SUB	Non utilisé.
1B	27	<CTRL><[>	<ESC>	ESC	Non utilisé en mode EDITION.
1C	28	<CTRL><\>	< → >	FS	Déplacement du curseur à droite.
1D	29	<CTRL><]>	< ← >	GS	Déplacement du curseur à gauche.
1E	30	<CTRL><^>	< ↑ >	RS	Déplacement du curseur en haut.
1F	31	<CTRL><->	< ↓ >	US	Déplacement du curseur en bas.
7F	127		<DEL>	DEL	Mode DESTRUCTION de l'éditeur.

Les codes de contrôle peuvent être accédés sous la forme codée par :

PRINT CHR$(n) en décimal ;

ou : PRINT CHR$(&Hn) en hexadécimal.

Caractères alphanumériques + caractères semi-graphiques (00H-FFH) (0D-255D)

BLANK (Space) — BLANK (NULL) — QPS = n_1 → — QMS = n_2 ↓

DEC	HEX	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0				Ø	@	P	`	p	Ç	É	á	Ã			α	≡
1	1	☺		!	1	A	Q	a	q	ü	æ	í	ã			β	±
2	2	☻		"	2	B	R	b	r	é	Æ	ô	Ĩ			Γ	≥
3	3	♥		#	3	C	S	c	s	â	ô	ú	ĩ			Π	≤
4	4	♦		$	4	D	T	d	t	ä	ö	ñ	Õ			Σ	⌠
5	5	♣		%	5	E	U	e	u	à	ò	Ñ	õ			σ	⌡
6	6	♠		&	6	F	V	f	v	å	û	a̲	Ũ			μ	÷
7	7	•		'	7	G	W	g	w	ç	ù	o̲	ũ			γ	≈
8	8	◘		(	8	H	X	h	x	ê	ÿ	¿	Ĳ		Δ	Φ	○
9	9	○		)	9	I	Y	i	y	ë	Ö	⌐	ĳ		ǂ	Θ	●
10	A	◙		*	:	J	Z	j	z	è	Ü	¬	¾		ω	Ω	•
11	B	♂		+	;	K	[	k	{	ï	¢	½	∽			δ	√
12	C	♀		,	<	L	\	l	¦	î	£	¼	◇			∞	ⁿ
13	D	♪		–	=	M	]	m	}	ì	¥	¡	‰			φ	²
14	E	♫		.	>	N	^	n	~	Ä	Pt	«	¶			∈	▯
15	F	☼		/	?	O	_	o		Å	ƒ	»	§			∩	

?CHR$(1) préalable (SOH) — BLANK (DEL) — BLANK(FF)

QPS → Quartet (le) plus significatif ou ½ octet supérieur.

QMS → Quartet (le) moins significatif ou ½ octet inférieur

Par accès programme (Logiciel)

- *En hexadécimal* : affichage direct à l'écran par : (HEX)

1- pour $n_1 n_2 \geq 20$ → PRINT CHR$(&Hn$_1n_2$)
2- pour $00 \leq n_1 n_2 \leq 1F$ → PRINT CHR$(1); CHR$(&Hn$_1$n$_2$+&H40)

- *En décimal* : affichage à l'écran par : (DEC)
(avec n = n1x16+n2)

1- pour n≥32 PRINT CHR$(n)
2- pour 0≤n≤31 PRINT CHR$(1); CHR$(n+64)

Remarque : CHR$(1) = en-tête de caractère semi-graphique compris entre 00H-1FH ou 0D-31D.

Par utilisation directe des touches (Matériel)

Voir codes accès clavier : NORMAL, NORMAL + <SHIFT>
<CODE>,<CODE> + <SHIFT>
<GRAPH>,<GRAPH> + <SHIFT>

Remarque : tous les caractères semi-graphiques doivent être utilisés en mode SCREEN 1 (matrice 8x8) pour éviter les troncatures.

BASIC

Tableau général des codes caractères (valeurs hexadécimales)

	MODE		0	1	2	3	4	5	6	7
0	Normal		0 30	1 31	2 32	3 33	4 34	5 35	6 36	7 37
		Shift	) 29	! 21	@ 4Ø	# 23	$ 24	% 25	^ 5E	& 26
	Graph		○ Ø9	¼ AC	½ AB	¾ BA	∩ EF	‰ BD	⌠ F4	√ FB
		Shift	◘ ØA		² FD	ⁿ FC			⌡ F5	
	Code		δ EB	ƒ 9F	‡ D9	§ BF	¢ 9B	ÿ 98	α EØ	β E1
		Shift	Δ 08	¡ AD	Pt 9E	¶ BE	£ 9C	¥ 9D		
1	Normal		8 38	9 39	- 2D	= 3D	\ 5C	[5B	] 5D	; 3B
		Shift	* 2A	(28	_ 5F	+ 2B	¦ 7C	{ 7B	} 7D	: 3A
	Graph		∞ EC	· Ø7	- 17	± F1	\ 1E	☺ Ø1	♪ ØD	♠ Ø6
		Shift		◙ Ø8	+ 1F	≡ FØ	¦ 16	☻ Ø2	♫ ØE	♦ Ø4
	Code		γ E7	ç 87	ε EE	θ E9	` 6Ø	φ ED	ω DA	ū 87
		Shift	Γ E2	Ç 8Ø				Φ E8	Ω EA	Ū 86
2	Normal		´ 27	£ 9C	, 2C	. 2E	/ 2F	FF	a 61	b 62
		Shift	" 22	~ 7E	< 3C	> 3E	? 3F	FF	A 41	B 42
	Graph		♣ Ø5	^ BB	≤ F3	≥ F2	/ 1D	FF	- C4	┴ 11
		Shift	♥ Ø3	≈ F7	« AE	» AF	÷ F6	FF	▌ FE	
	Code		ĳ B9	σ E5	å 86	ª A6	º A7	FF	ä 84	ù 97
		Shift	Ĳ B8	Σ E4	Å 8F		¿ A8	FF	Ä 8E	
3	Normal		c 63	d 64	e 65	f 66	g 67	h 68	i 69	j 6A
		Shift	C 43	D 44	E 45	F 46	G 47	H 48	I 49	J 4A
	Graph		◊ BC	C7	CD	├ 14	┼ 15	┤ 13	DC	C6
		Shift	- FA	C1	CE	D4	┼ 1Ø	D6	DF	CA
	Code		ı 8D	ï 8B	î 8C	ö 94	ü 81	ã B1	í A1	œ 91
		Shift				Ö 99	Ü 9A	Ã 8Ø		Æ 92
4	Normal		k 6B	l 6C	m 6D	n 6E	o 6F	p 7Ø	q 71	r 72
		Shift	K 4B	L 4C	M 4D	N 4E	O 4F	P 5Ø	Q 51	R 52
	Graph		DD	C8	♂ ØB	1B	C2	█ DB	╲ CC	18
		Shift	DE	C9	♀ ØC	D3	C3	D7	╱ CB	⌐ A9
	Code		ĩ B3	õ B5	µ E6	ñ A4	ó A2	ú A3	â 83	ô 93
		Shift	Ĩ B2	Õ B4		Ñ A5		π E3		
5	Normal		s 73	t 74	u 75	v 76	w 77	x 78	y 79	z 7A
		Shift	S 53	T 54	U 55	V 56	W 57	X 58	Y 59	Z 5A
	Graph		D2	┬ 12	CØ	1A	CF	╳ 1C	19	☼ ØF
		Shift	D1		C5	D5	DØ	• F9	¬ AA	° F8
	Code		ë 89	û 96	é 82	ò 95	ê 88	è 8A	á AØ	à 85
		Shift			É 90					

Classement par ordre alphabétique des mots-clés

Mot-clé	*Token DEC*	*Token HEX*	*Mot-clé*	*Token DEC*	*Token HEX*
*	243	F3	+	241	F1
-	242	F2	/	244	F4
<	240	F0	=	239	EF
>	238	EE	ABS	134*	86*
AND	246	F6	ASC	149*	95*
ATN	142*	8E*	ATTR$	233	E9
AUTO	169	A9	BASE	201	C9
BEEP	192	C0	BIN$	157*	9D*
BLOAD	207	CF	BSAVE	208	D0
CALL	202	CA	CDBL	160*	A0*
CHR$	150*	96*	CINT	158*	9E*
CIRCLE	188	BC	CLEAR	146	92
CLOAD	155	9B	CLOSE	180	B4
CLS	159	9F	CMD	215	D7
COLOR	189	BD	CONT	153	99
COPY	214	D6	COS	140*	8C*
CSAVE	154	9A	CSNG	159*	9F*
CSRLIN	232	E8	CVD	170*	AA*
CVI	168*	A8*	CVS	169*	A9*
DATA	132	84	DEF	151	97
DEFDBL	174	AE	DEFINT	172	AC
DEFSNG	173	AD	DEFSTR	171	AB
DELETE	168	A8	DIM	134	86
DRAW	190	BE	DSKF	166*	A6*
DSKI$	234	EA	DSKO$	209	D1
ELSE	161	A1	END	129	81
EOF	171	AB*	EQV	249	F9
ERASE	165	A5	ERL	225	E1
ERR	226	E2	ERROR	166	A6
EXP	139	8B*	FIELD	177	B1
FILES	183	B7	FIX	161*	A1*
FN	222	DE	FOR	130	82
FPOS	167*	A7*	FRE	143*	8F*
GET	178	B2	GOSUB	141	8D
GOTO	137	89	HEX$	155*	9B*
IF	139	8B	IMP	250	FA
INKEY$	236	EC	INP	144	90*
INPUT	133	85	INSTR	229	E5
INT	133*	85*	IPL	213	D5
KEY	204	CC	KILL	212	D4

* *Les tokens avec astérisque sont les tokens à deux octets. Seul le 2e est indiqué, le 1er étant toujours 255D=FFH.*

Mot-clé	Token DEC	Token HEX	Mot-clé	Token DEC	Token HEX
LEFT$	129*	81*	LEN	146*	92*
LET	136	88	LFILES	187	BB
LINE	175	AF	LIST	147	93
LLIST	158	9E	LOAD	181	B5
LOC	172*	AC*	LOCATE	216	D8
LOF	173	AD*	LOG	138	8A*
LPOS	156*	9C*	LPRINT	157	9D
LSET	184	B8	MAX	205	CD
MERGE	182	B6	MID$	131*	83*
MKD$	176*	B0*	MKI$	174*	AE*
MKS$	175*	AF*	MOD	251	FB
MOTOR	206	CE	NAME	211	D3
NEW	148	94	NEXT	131	83
NOT	224	E0	OCT$	154*	9A*
OFF	235	EB	ON	149	95
OPEN	176	B0	OR	247	F7
OUT	156	9C	PAD	165	A5*
PAINT	191	BF	PDL	164*	A4*
PEEK	151*	97*	PLAY	193	C1
POINT	237	ED	POKE	152	98
POS	145*	91*	PRESET	195	C3
PRINT	145	91	PSET	194	C2
PUT	179	B3	READ	135	87
REM	143	8F	RENUM	170	AA
RESTORE	140	8C	RESUME	167	A7
RETURN	142	8E	RIGHT$	130*	82*
RND	136*	88*	RSET	185	B9
RUN	138	8A	SAVE	186	BA
SCREEN	197	C5	SET	210	D2
SGN	132*	84*	SIN	137*	89*
SOUND	196	C4	SPACE$	153*	99*
SPC(	223	DF	SPRITE	199	C7
SQR	135*	87*	STEP	220	DC
STICK	162	A2*	STOP	144	90
STR$	147*	93*	STRIG	163*	A3*
STRING$	227	E3	SWAP	164	A4
TAB(	219	DB	TAN	141*	8D*
THEN	218	DA	TIME	203	CB
TO	217	D9	TROFF	163	A3
TRON	162	A2	USING	228	E4

* Les tokens avec astérisque sont les tokens à deux octets. Seul le 2e est indiqué, le 1er étant toujours 255D=FFH.

Mot-clé	Token DEC	Token HEX	Mot-clé	Token DEC	Token HEX
USR	221	DD	VAL	148*	94*
VARPTR	231	E7	VDP	200	C8
VPEEK	152*	98*	VPOKE	198	C6
WAIT	150	96	WIDTH	160	A0
XOR	248	F8	\	252	FC
^	245	F5			

Classement par numéro de token :
mots-clés codés sur un octet

Token DEC	Token HEX	Mot-clé	Token DEC	Token HEX	Mot-clé
129	81	END	130	82	FOR
131	83	NEXT	132	84	DATA
133	85	INPUT	134	86	DIM
135	87	READ	136	88	LET
137	89	GOTO	138	8A	RUN
139	8B	IF	140	8C	RESTORE
141	8D	GOSUB	142	8E	RETURN
143	8F	REM	144	90	STOP
145	91	PRINT	146	92	CLEAR
147	93	LIST	148	94	NEW
149	95	ON	150	96	WAIT
151	97	DEF	152	98	POKE
153	99	CONT	154	9A	CSAVE
155	9B	CLOAD	156	9C	OUT
157	9D	LPRINT	158	9E	LLIST
159	9F	CLS	160	A0	WIDTH
161	A1	ELSE	162	A2	TRON
163	A3	TROFF	164	A4	SWAP
165	A5	ERASE	166	A6	ERROR
167	A7	RESUME	168	A8	DELETE
169	A9	AUTO	170	AA	RENUM
171	AB	DEFSTR	172	AC	DEFINT
173	AD	DEFSNG	174	AE	DEFDBL
175	AF	LINE	176	B0	OPEN
177	B1	FIELD	178	B2	GET
179	B3	PUT	180	B4	CLOSE
181	B5	LOAD	182	B6	MERGE
183	B7	FILES	184	B8	LSET

* Les tokens avec astérisque sont les tokens à deux octets. Seul le 2e est indiqué, le 1er étant toujours 255D=FFH.

Token DEC	Token HEX	Mot-clé	Token DEC	Token HEX	Mot-clé
185	B9	RSET	186	BA	SAVE
187	BB	LFILES	188	BC	CIRCLE
189	BD	COLOR	190	BE	DRAW
191	BF	PAINT	192	C0	BEEP
193	C1	PLAY	194	C2	PSET
195	C3	PRESET	196	C4	SOUND
197	C5	SCREEN	198	C6	VPOKE
199	C7	SPRITE	200	C8	VDP
201	C9	BASE	202	CA	CALL
203	CB	TIME	204	CC	KEY
205	CD	MAX	206	CE	MOTOR
207	CF	BLOAD	208	D0	BSAVE
209	D1	DSKO$	210	D2	SET
211	D3	NAME	212	D4	KILL
213	D5	IPL	214	D6	COPY
215	D7	CMD	216	D8	LOCATE
217	D9	TO	218	DA	THEN
219	DB	TAB(	220	DC	STEP
221	DD	USR	222	DE	FN
223	DF	SPC(	224	E0	NOT
225	E1	ERL	226	E2	ERR
227	E3	STRING$	228	E4	USING
229	E5	INSTR	230	E6	' (REM)
231	E7	VARPTR	232	E8	CSRLIN
233	E9	ATTR$	234	EA	DSKI$
235	EB	OFF	236	EC	INKEY$
237	ED	POINT	238	EE	>
239	EF	=	240	F0	<
241	F1	+	242	F2	-
243	F3	*	244	F4	/
245	F5	^	246	F6	AND
247	F7	OR	248	F8	XOR
249	F9	EQV	250	FA	IMP
251	FB	MOD	252	FC	\

Classement par numéro de token : mots-clés codés sur deux octets (le premier octet est toujours FF-255)

Token DEC	Token HEX	Mot-clé	Token DEC	Token HEX	Mot-clé
129	81	LEFT$	153	99	SPACE$
130	82	RIGHT$	154	9A	OCT$
131	83	MID$	155	9B	HEX$
132	84	SGN	156	9C	LPOS
133	85	INT	157	9D	BIN$
134	86	ABS	158	9E	CINT
135	87	SQR	159	9F	CSNG
136	88	RND	160	A0	CDBL
137	89	SIN	161	A1	FIX
138	8A	LOG	162	A2	STICK
139	8B	EXP	163	A3	STRIG
140	8C	COS	164	A4	PDL
141	8D	TAN	165	A5	PAD
142	8E	ATN	166	A6	DSKF
143	8F	FRE	167	A7	FPOS
144	90	INP	168	A8	CVI
145	91	POS	169	A9	CVS
146	92	LEN	170	AA	CVD
147	93	STR$	171	AB	EOF
148	94	VAL	172	AC	LOC
149	95	ASC	173	AD	LOF
150	96	CHR$	174	AE	MKI$
151	97	PEEK	175	AF	MKS$
152	98	VPEEK	176	B0	MKD$

BASIC

Toute erreur pouvant survenir dans un programme (ou en mode direct), qu'elle soit due à sa rédaction (syntaxe ou logique erronée, instructions manquantes), ou produite par l'exécution elle-même (dépassement de capacité, etc.) est traitée par le MSX-Basic. Celui-ci interrompt alors l'exécution et affiche un message d'erreur approprié selon la syntaxe suivante :

XXXX in [Nx de ligne]

dans laquelle **XXXX** est un message définissant le type de l'erreur dont la liste est donnée ci-dessous et **[Nx de ligne]** est la ligne où l'erreur s'est produite.

Parallèlement, si la commande "ON ERROR GOTO" figure au début du programme, les variables réservées ERR et ERL contiendront respectivement le numéro de l'erreur et le numéro de la ligne où l'erreur s'est produite.

N° code	*Message/diagnostic*
1	**NEXT without FOR** Une variable appartenant à une instruction NEXT ne correspond à aucune variable d'une instruction FOR préalablement épellée, ponctuation incorrecte, etc). condition.
2	**Syntax error** Une séquence incorrecte de caractères est rencontrée dans une ligne (nombre impair de parenthèses, commande ou instruction mal épelée, ponctuation incorrecte, etc).
3	**RETURN without GOSUB** Une instruction RETURN est rencontrée, sans GOSUB préalable.
4	**Out of DATA** Une instruction READ est rencontrée alors qu'il n'y a plus d'instruction DATA avec des données non lues.
5	**Illegal function call** Un paramètre hors limites est envoyé à une fonction mathématique ou alphanumérique. Ce message peut aussi se produire dans les circonstances suivantes : - indice négatif ou démesuré ; - argument négatif ou nul avec LOG ou négatif avec SQR ; - usage de USR alors que l'adresse de début n'a pas été définie ; - usage de ERASE, SWAP ou VARPTR avec une variable non identifiée (non utilisée) ; - argument impropre pour MID$, LEFT$, RIGHT$, INP, OUT, WAIT, PEEK, POKE, TAB, SPC, STRING$, SPACE$, INSTR, ON...GOTO (GOSUB), etc. ;

N° code	*Message/diagnostic*
	- numéro d'enregistrement négatif pour GET ou PUT ; - instruction graphique en SCREEN 0 ou 1 ; - caractère interdit en MLG (Macro langage graphique) ou en MLM (Macro langage musical) ; - usage de SPRITE en SCREEN 0.
6	**Overflow** Le résultat est trop grand pour être représenté en format numérique Basic. Quand un "Underflow" se produit, le résultat est posé égal à zéro et l'exécution se poursuit sans donner de message d'erreur.
7	**Out of memory** Un programme est trop long, contient trop de boucles FOR, de GOSUB ou de variables.
8	**Undefined line number** Appel à une ligne inexistante dans un GOTO, GOSUB, IF...THEN...ELSE ou DELETE. Un nouveau numéro de ligne entré sans texte produit également cette erreur.
9	**Subscript out of range** Appel à un élément de tableau avec un indice hors limites ou avec un nombre incorrect d'indices.
10	**Redimensioned array** Deux instructions DIM sont données pour le même tableau ou bien une instruction DIM est donnée après que ce tableau ait reçu, par défaut, la dimension 10.
11	**Division by zéro** Division par zéro ou bien expression nulle élevée à une puissance négative.
12	**Illegal direct** Une instruction qui ne peut que se trouver dans un programme est entrée en mode direct.
13	**Type mismatch** Une variable alphanumérique reçoit une valeur numérique ou vice versa ; une fonction qui attend un argument numérique reçoit un argument alphanumérique ou vice versa.
14	**Out of string space** Les variables alphanumériques forcent le Basic à dépasser la mémoire de l'espace réservé pour chaînes qui a été défini par une instruction CLEAR.
15	**String too long** Tentative de création d'une chaîne de plus de 255 caractères.

BASIC

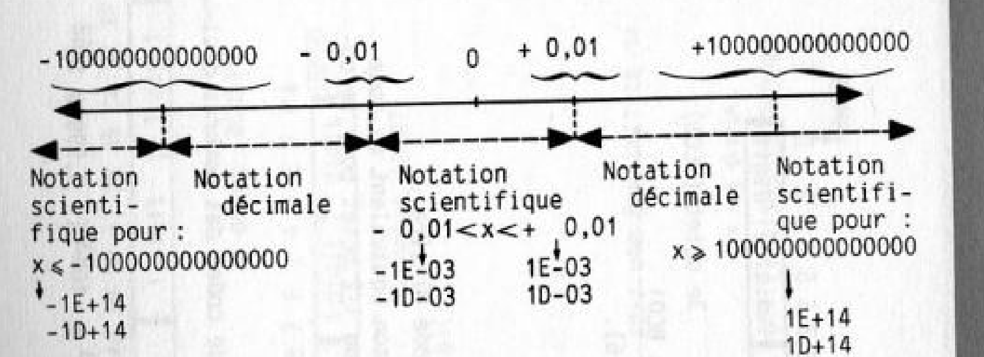
N° code	Message/diagnostic
16	**String formula too complex** L'expression d'une chaîne est trop longue ou trop complexe et devrait être fractionnée en expressions plus simples.
17	**Can't continue** Tentative de relancer l'exécution d'un programme qui : - a été interrompu, suite à une erreur ; - a été modifié durant une interruption ; - n'existe pas.
18	**Undefined user function** Un appel de la fonction USR a lieu avant que la définition (DEF) ait été donnée.
19	**Device I/O error** Une erreur se produit lors d'une opération de dispositif I/O.
20	**Verify error** Le contenu de la mémoire et celui d'un fichier se révèlent différents. Cette erreur peut se produire lors de l'exécution d'une commande CLOAD ?
21	**No RESUME** Une routine de traitement d'erreur est engagée mais ne contient pas d'instruction RESUME.
22	**RESUME without error** Une instruction RESUME est rencontrée avant l'appel d'une routine de traitement d'erreur.
23	**Unprintable error** Aucun message d'erreur n'est disponible pour décrire la situation existante.
24	**Missing operand** Une expression contient un opérateur non suivi d'opérande ; une commande/instruction est donnée sans les paramètres nécessaires.
25	**Line buffer overflow** Tentative d'entrer (INPUT) une ligne qui contient plus de 255 caractères.
26~49	**Unprintable error** Aucun message d'erreur n'est disponible pour décrire la situation existante.

N° code	Message/diagnostic
50*	**Field overflow** Une instruction FIELD tente d'accorder plus d'octets qu'il n'a été spécifié pour la longueur d'un enregistrement de fichier à accès sélectif.
51	**Internal error** Un mauvais fonctionnement interne s'est produit en MSX-BASIC.
52	**Bad file number** Une instruction/commande se refère à un numéro de fichier qui n'a pas été ouvert ou qui est hors des limites des numéros de fichier fixées lors de l'initialisation.
53*	**File not found** Une instruction/commande LOAD, KILL, NAME ou OPEN concerne un fichier qui n'existe pas sur le disque courant.
54	**File already open** Une instruction OPEN pour sortie séquentielle est émise pour un fichier déjà ouvert ; une instruction KILL concerne un fichier ouvert.
55	**Input past end** Une instruction INPUT est exécutée après que toutes les données d'un fichier aient été entrées, ou pour un fichier vide. Pour éviter cette erreur, détectez la fin du fichier avec la fonction EOF.
56	**Bad file name** Utilisation d'un nom de fichier interdit avec un LOAD, KILL ou OPEN (nom comprenant trop de caractères).
57	**Direct statement in file** Une commande à n'utiliser qu'en mode direct est rencontrée lors d'un chargement (LOAD) en format ASCII. L'action du LOAD prend fin.
58*	**Sequential I/O only** Utilisation d'un GET ou un PUT alors que le fichier concerné est ouvert en tant que fichier séquentiel.
59	**File not OPEN** Une commande/instruction I/O est utilisée pour un fichier qui n'a pas été ouvert.
60*	**Bad FAT** La table d'attribution de fichier (FAT) n'est pas en ordre. Le disque n'a probablement pas été initialisé par la commande FORMAT.

B A S I C

N° code	Message/diagnostic
61*	**Bad file mode** Tentative d'utilisation de PUT, GET ou LOF avec un fichier séquentiel, d'utiliser un LOAD avec un fichier à accès sélectif ou encore d'exécuter un OPEN avec un mode de fichier incorrect.
62*	**Bad drive name** Utilisation d'un nom de drive interdit.
63*	**Bad sector number** Mauvais numéro de secteur.
64*	**File still open** Un fichier n'est pas encore fermé.
65	**File already exists** Le nom de fichier mentionné dans une instruction NAME est identique à un nom de fichier déjà présent sur le disque.
66*	**Disk full** Tout l'espace du disque a été utilisé.
67*	**Too many files** Tentative de création d'un nouveau fichier (à l'aide de SAVE ou OPEN) sur un disque qui contient déjà 255 fichiers.
68*	**Disk write protected** Le disque est muni d'une languette de protection à l'écriture.
69*	**Disk I/O error** Une erreur irrémédiable s'est produite durant une opération I/O avec disque.
70	**Disk offline** Le drive n'est pas en ligne.
71	**Rename across disks** Tentative de donner un nouveau nom à un fichier en se trompant sur le nom du drive.
72~255	**Unprintable error** Aucun message d'erreur n'est disponible pour décrire la situation existante.

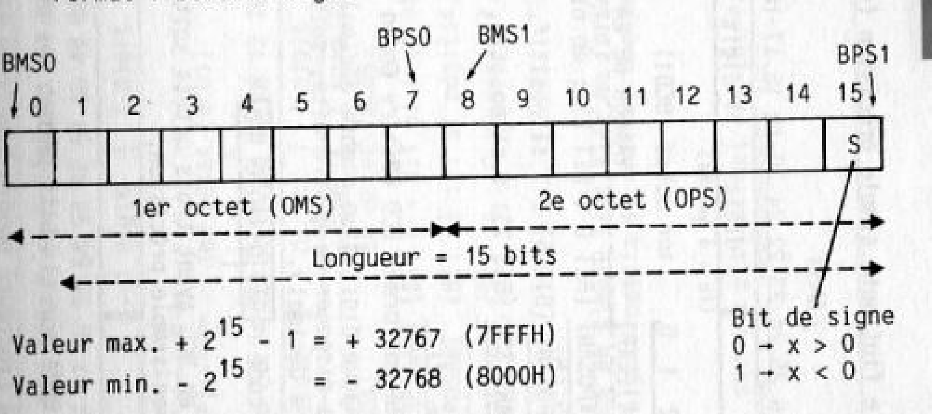
* *concerne spécifiquement l'extension DISQUE du MSX-BASIC.*



FORMAT DE STOCKAGE EN MEMOIRE DES NOMBRES SIGNES

1 - <u>Nombre entier</u> (2 octets)

Format : binaire signé



Valeur max. + 2^{15} - 1 = + 32767 (7FFFH)

Valeur min. - 2^{15} = - 32768 (8000H)

Nombres > 0 → N DEC = OMS x 256 x OPS.

Les nombres < 0 sont obtenus par complément à 2 des nombres > 0 signés.

Méthode : 1- Inverser tous les bits
2- Ajouter 1.

2 - Nombre en virgule flottante simple précision (4 octets) : N = M.10^E

BPS3 ↓ BMS3 ↓ BMS0 ↓

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

| S1 | S2 | | 1 digit | 1 digit | 1 digit | 1 digit | 1 digit | 1 digit |

(le + à gauche) (le + à droite)

7 6 5 4 3 2 1 0 — 1er octet (BCD1) — 2e octet (BCD2) — 3e octet (BCD3)

EXPOSANT (1 octet)(EXP) — VALEUR DE LA MANTISSE (3 octets en BCD)

→ VALEUR DE L'EXPOSANT (BIT 0 - BIT 5) : en binaire signé (avec bit 6).
→ SIGNE DE L'EXPOSANT (BIT 6) : 1 si positif, 0 si négatif.
→ SIGNE DE LA MANTISSE (BIT 7): 0 si positif, 1 si négatif.

Valeur de la mantisse (M)

Elle est représentée en format BCD (**B**inary **C**oded **D**ecimal) ou (**D**écimal **C**odé **B**inaire).

Dans ce système, chaque digit d'un nombre décimal est représentant par son équivalent en code binaire (nécessite de disposer de quatre bits, soit un digit hexadécimal ou 1/2 octet pour le codage de chaque chiffre décimal).

Exemple : 12 D est codé | 0 0 0 1 | 0 0 1 0 | (1, 2)

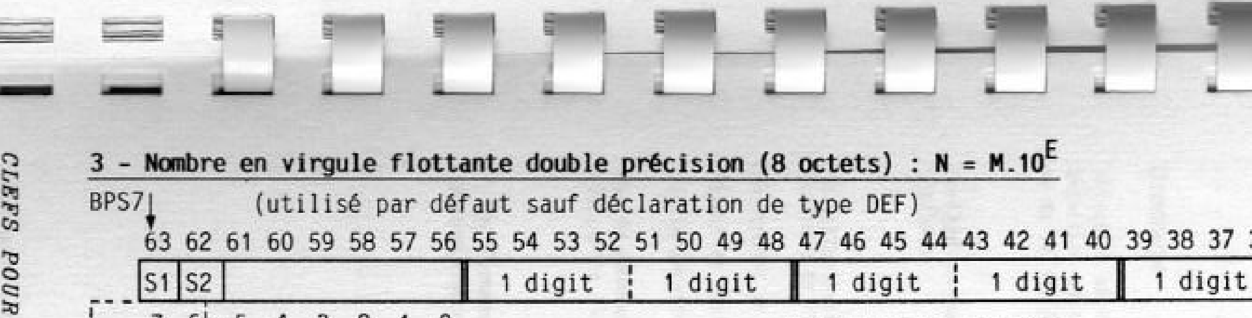
La mantisse, codée en BCD ayant trois octets soit six quartets, permet de coder des nombres décimaux de six chiffres en simple précision.

Valeur maximale

L'exposant décimal codé sur six bits permet des chiffres allant de $+10^{63}$ à -10^{63} (+ 9,99999 10^{62} à - 9,99999 10^{62}).

3 - Nombre en virgule flottante double précision (8 octets) : N = M.10^E

BPS7 ↓ (utilisé par défaut sauf déclaration de type DEF)



→ VALEUR DE L'EXPOSANT (BIT 0 - BIT 5) : en binaire signé (avec bit 6).
→ SIGNE DE L'EXPOSANT (BIT 6) : 1 si positif, 0 si négatif.
→ SIGNE DE LA MANTISSE (BIT 7) : 0 si positif, 1 si négatif.

Valeur de la mantisse (M)

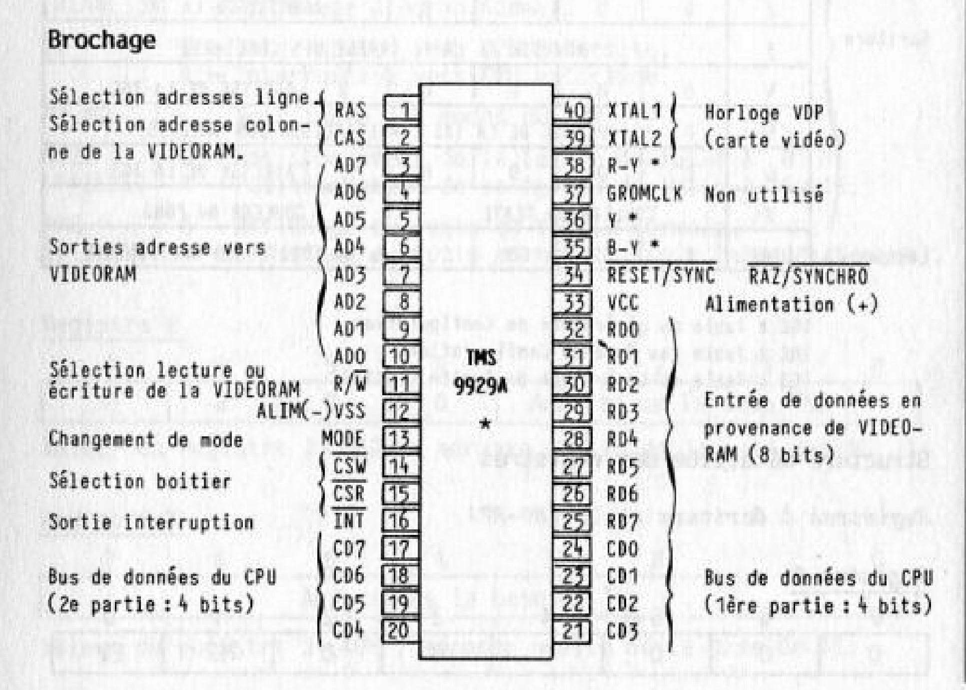
Elle est codée comme en simple précision en format BCD sur sept octets, soit une possibilité de 2 x 7 = 14 chiffres significatifs (1 chiffre décimal par demi-octet).

Valeur maximale

L'exposant décimal sur six bits permet des chiffres allant de $+10^{63}$ à -10^{63} (+ 9,9999999999999 (+ 9,9999999999999 10^{62} à -9,9999999999999 10^{62}).

PROCESSEUR GRAPHIQUE (VDP)

Brochage



** En **Europe uniquement** : 625 lignes/sortie des signaux de chrominance R-Y et R-Y et luminance Y.*

Structure des registres internes
Tableau récapitulatif

	REG.	B7	B6	B5	B4	B3	B2	B1	B0
Ecriture	0	0	0	0	0	0	0	M3	EV
Ecriture	1	1	BLANC	IE	M1	M2	0	TAILLE	F.AGR
Ecriture	2	0	0	0	0	ADRESSE DE LA TNC			
Ecriture	3	ADRESSE DE LA TC (TABLE DES COULEURS)							
Ecriture	4	0	0	0	0	0	ADRESSE DE LA TGC		
Ecriture	5	0	ADRESSE DE LA TAS (TABLE ALLOCATION SPRITES)						
Ecriture	6	0	0	0	0	0	ADRESSE DE LA TGS		
Ecriture	7	COULEUR DU TEXTE				COULEUR DU FOND			
Lecture →	ETAT	F	5S	COL	N° du 5e SPRITE SUR HORIZONTALE				

TGC : Table du générateur de Configuration.
TNC : Table des Noms de Configuration.
TGS : Table du Générateur de Sprite (lutin).

Structure détaillée des registres

Registres à écriture seule (R0-R7)

Registre 0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	M3	EV

M3 Voir table des modes du VDP.

EV : 0 → Signal d'entrée VIDEO externe non autorisé.
1 → Signal d'entrée VIDEO externe autorisé.

Registre 1

7	6	5	4	3	2	1	0
4/16K	BLANK	IE	M1	M2	0	SIZE	MAG

4/16K : 0 → RAM dynamique de 4K x 1 bit utilisé comme VRAM.
1 → RAM dynamique de 16K utilisé comme VRAM.

BLANK (BLANC) : 0 → Arrêt de l'affichage écran.
1 → Affichage d'écran normal.

IE : 0 → Interruption vers CPU interdite.
1 → Interruption vers CPU autorisée.

M1,M2 : Voir table des modes du VDP.

SIZE (TAILLE) : 0 → Positionnement de la taille du lutin à 8 x 8.
1 → Positionnement de la taille du lutin à 16 x 16.

MAG (F.AGR) : 0 → Affichage du lutin en taille normale.
1 → Affichage du lutin en mode élargie (agrandissement).

Registre 2

7	6	5	4	3	2	1	0
0	0	0	0	Adresse de la base TNC			

Valeur du registre 2 x 400H : adresse réelle de la base de TNC.

Registre 3

7	6	5	4	3	2	1	0
Adresse de la base de TC							

Valeur du registre 3 x 40H : adresse réelle de la base de TC.

Registre 4

7	6	5	4	3	2	1	0
0	0	0	0	0	Adresse de la base TGC		

Valeur du registre 4 x 800H : adresse réelle de la base de la TGC.

Registre 5

7	6	5	4	3	2	1	0
0	Adresse de la base de la TAS						

Valeur du registre 5 x 80H : adresse réelle de la base de la TAS.

Registre 6

7	6	5	4	3	2	1	0
0	0	0	0	0	Adresse de la base de TGS		

Valeur du registre 6 x 800H : adresse réelle de la base de TGS.

Registre 7

7	6	5	4	3	2	1	0
Couleur caractère				Couleur de fond			

Registre à lecture seule (R8)
(registre d'état)

Registre 8

7	6	5	4	3	2	1	0
F	5S	C	5SN				

F : **drapeau d'interruption** mis à 1 à la fin d'un affichage écran, remis à zéro après que le registre d'état soit lu ou lorsque VDP est remis à zéro extérieurement.

5S : **drapeau de 5e Sprite** : mis à 1 quand il y a plus de quatre Sprites sur une ligne horizontale (0 à 192).

C : **drapeau de collision de Sprites** : le VDP vérifie qu'au moins deux Sprites se recouvrent sur plus d'un pixel (ce contrôle est effectué tous les 1/50 de secondes).

5SN : (BIT0-BIT4) - **Numéro de 5e Sprite**. S'il y a plus de quatre Sprites sur une ligne horizontale et si 5S est mis à 1, alors le numéro du Sprite violant le 5e Sprite est donné par ce **numéro**.

Table des modes de fonctionnement de VDP

Mode d'écran		*M1*	*M2*	*M3*
40 x 24 (mode texte)	(mode 0)	1	0	0
32 x 24 (mode texte ou graphique 1)	(mode 1)	0	0	0
Mode haute résolution graphique 2	(mode 2)	0	0	1
Mode multicolore	(mode 3)	0	1	0

M1, M2 : Bit 1 + bit 3 Registre 1 — M3 : Bit 1 Reg.0

Table des codes de couleur (utilisée par registre 7)

Code DEC	*Code HEX*	*Couleur*	*Code DEC*	*Code HEX*	*Couleur*
0	0	Transparent	8	8	Rouge moyen
1	1	Noir	9	9	Rouge clair
2	2	Vert moyen	10	A	Jaune foncé
3	3	Vert clair	11	B	Jaune clair
4	4	Bleu foncé	12	C	Vert foncé
5	5	Bleu clair	13	D	Magenta
6	6	Rouge foncé	14	E	Gris
7	7	Cyan	15	F	Blanc

Cyan : bleu + vert
Magenta : bleu + rouge
Jaune : rouge + vert

Adresses standards des tables

Mode	*TNC*	*TGC*	*TC*	*TAS*	*TGS*
Texte	0000H	0800H	-	-	-
Graphique 1+2	1800H	0000H	2000H	1B00H	3800H
Multicolore	0800H	0000H	-	1B00H	3800H

avec :

0000H ⩽ VIDEORAM ⩽ 3FFFH.

Port	Bit	E/S	Nom du signal	Rôle
A	0	SORTIE	CS0L	Signal de sélection de "fente"* pour la page mémoire n° 1.
	1		CS0H	Adresses : 0000H à 3FFFH.
	2		CS1L	Signal de sélection de "fente"* pour la page mémoire n° 2.
	3		CS1H	Adresses : 4000H à 7FFFH.
	4		CS2L	Signal de sélection de "fente"* pour la page mémoire n° 3.
	5		CS2H	Adresses : 8000H à BFFFH.
	6		CS3L	Signal de sélection de "fente"* pour la page mémoire n° 4.
	7		CS3H	Adresses : C000H à FFFFH.
B	0 à 7	ENTREE		Lecture de la matrice clavier Signal de retour clavier
C	0 1 2 3	SORTIE	KB0 KB1 KB2 KB3	Signal de scrutation clavier Ecriture de la ligne clavier (0-9)
	4		CASON	Démarrage moteur Signal de contrôle cassette } actif niveau bas (0)
	5		CASW	Signal d'écriture cassette.
	6		CAPS	Actif niveau bas (0) Signal de commande d'affichage lampe CAPS
	7		SOUND	Entrée "SOUND" par logiciel (1 bit).

bits n° 1, 3, 5, 7
bits n° 0, 2, 4, 6

* Fente 0 → 00
Fente 1 → 01
Fente 2 → 10
Fente 3 → 11

Exemple : page 0 fente 0, page 2 fente 1
page 1 fente 0, page 3 fente 2

7	6	5	4	3	2	1	0
1	0	0	1	0	0	0	0

La fente 0 (SLOT 0) est celle représentée par le système de base.

Contenu des registres internes

Reg	Utilisation	B7	B6	B5	B4	B3	B2	B1	B0	Plage réglage
R0	F. CANAL A	8 BITS DE REGLAGE FIN DE LA FREQUENCE CANAL A								0-255
R1	F. CANAL A	////////////////////				REGL.GROSSIER CANAL A				0-15
R2	F. CANAL B	8 BITS DE REGLAGE FIN DE LA FREQUENCE CANAL B								0-255
R3	F. CANAL B	////////////////////				REGL.GROSSIER CANAL B				0-15
R4	F. CANAL C	8 BITS DE REGLAGE FIN DE LA FREQUENCE CANAL C								0-255
R5	F. CANAL C	////////////////////				REGL.GROSSIER CANAL C				0-15
R6	PERIODE BRUIT	///////////////			5 BITS DE CONTROLE PERIODE					0-31
R7	SELECTION	IN OUT		BRUIT			FREQUENCE			0-63
R8	AMPLITUDE A	///////////////			M	L3	L2	L1	L0	0-16
R9	AMPLITUDE B	///////////////			M	L3	L2	L1	L0	0-16
R10	AMPLITUDE C	///////////////			M	L3	L2	L1	L0	0-16
R11	PERIODE ENVP.	8 BITS REGLAGE FIN DE LA PERIODE D'ENVELOPPE								0-255
R12	PERIODE ENVP.	8 BITS REGLAGE GROSSIER PERIODE D'ENVELOPPE								0-255
R13	FORME ENVP.	////////////////////				CONT E3	ATT E2	ALT E1	HOLD E0	0-15
R14	PORT E/S A	8 BITS PORT PARALLELE A *								
R15	PORT E/S B	8 BITS PORT PARALLELE B **								

* Utilisé par JOYSTICK
B0 vers le haut
B1 vers le bas
B2 vers la gauche
B3 vers la droite
B4 bouton de tir
B5 trigger analogique
B6 inutilisé
B7 lecture de la cassette

** Utilisé par MANETTE ANALOGIQUE / JOYSTICK

B0 - B5 : manette analogique
B6 : section du Joystick 1 ou 2
B7 : inutilisé.

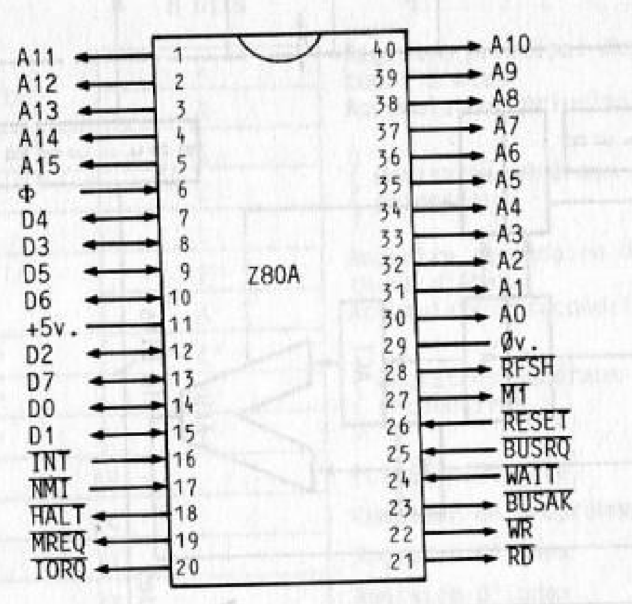
Forme des signaux générés (enveloppes)
(disponibles en R13)

E3	E2	E1	E0	Valeurs possibles DEC	
0	0	X	X	0.1.2.3	
0	1	X	X	4.5.6.7	
1	0	0	0	8	
1	0	0	1	9	
1	0	1	0	10	
1	0	1	1	11	
1	1	0	0	12	
1	1	0	1	13	
1	1	1	0	14	
1	1	1	1	15	

X = état indifférent.

MICROPROCESSEUR Z80

BROCHAGE

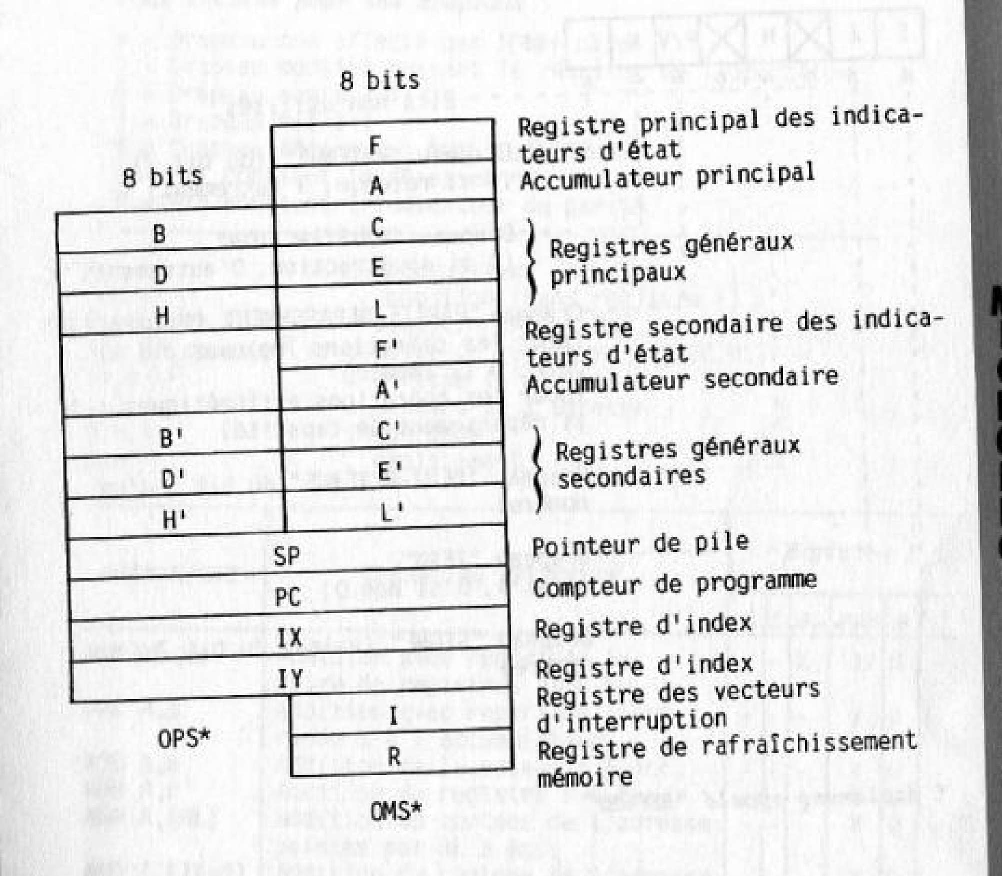


1-Bit d'adresse 11
2-Bit d'adresse 12
3-Bit d'adresse 13
4-Bit d'adresse 14
5-Bit d'adresse 15
6-Horloge
7-Bit de donnée 4
8-Bit de donnée 3
9-Bit de donnée 5
10-Bit de donnée 6
11-Tension de 5 volts régulés
12-Bit de donnée 2
13-Bit de donnée 7
14-Bit de donnée Ø
15-Bit de donnée 1
16-Interruptions masquables
17-Interruptions non masquables
18-Signal d'arrêt du microprocesseur
19-Demande d'opération mémoire
20-Demande d'entrées/sorties
21-Commande de lecture mémoire
22-Commande d'écriture mémoire
23-Acceptation d'accès direct mémoire
24-Demande d'attente au microprocesseur
25-Demande d'accès direct mémoire
26-Initialisation du microprocesseur
27-Signal de début de cycle
28-Rafraîchissement des mémoires dynamiques
29-Tension Ø volt masse électrique
30-Bit d'adresse Ø
31-Bit d'adresse 1
32-Bit d'adresse 2
33-Bit d'adresse 3
34-Bit d'adresse 4
35-Bit d'adresse 5
36-Bit d'adresse 6
37-Bit d'adresse 7
38-Bit d'adresse 8
39-Bit d'adresse 9
40-Bit d'adresse 1Ø

Z 80 A

BUFFER
BUFFER
BUFFER
Øv
+5v
Ø (horloge)
8 bits
16 bits
F
F'
A
A'
ACT
TMP
ALU
Bus de données
Bus d'adresses
Bus de commandes
MUX
MUX
H L
B C
D E
H' L'
B' C'
D' E'
IX
IY
SP
PC
+
+1
I
R
Registre d'instruction
Décodeur
Contrôleur-séquenceur

Structure générale



* dans le cas d'utilisation des registres par paires.

Structure détaillée du registre F

```
Bit 7                 Bit 0
┌─┬─┬─┬─┬─┬───┬─┬─┐
│S│Z│X│H│X│P/V│N│C│  (F)
└─┴─┴─┴─┴─┴───┴─┴─┘
```

- Bits non utilisés
- Drapeau "RETENUE"* (du bit 7) (1 si retenue, 0 autrement)
- Drapeau "SOUSTRACTION" (1 si soustraction, 0 autrement)
- Drapeau "PARITE/DEPASSEMENT DE CAPACITE" (pour les opérations logiques : 1 si pair, 0 si impair) (pour les opérations arithmétiques : 1 si dépassement de capacité)
- Drapeau "DEMI-RETENUE" (du bit 3 d'un nombre)
- Drapeau "ZERO" (1 si 0, 0 si non 0)
- Drapeau "SIGNE" (valeur du bit 7 d'un nombre)

* *également appelé "REPORT"*

Jeu d'instructions

Code utilisé pour les drapeaux

● = Drapeau non affecté par l'opération
\- = Drapeau modifié suivant le résultat de l'opération
0 = Drapeau remis à 0
1 = Drapeau mis à 1
X = Drapeau inconnu ou état indifférent
V = P/V contient le dépassement
P = P/V contient l'indicateur de parité

Acc	: Accumulateur (A)
cc	: condition (dans registre F)
Reg p,q,r,s	: registre (A,B,C,D,E,H,L)
Reg dd,pp,qq,rr,ss	: paire de registres (BC,DE,HL)
nn,p	: adresse (p utilisé avec RST)
( )	: contenu de l'adresse
m,n,s	: opérande
d,e	: déplacement

MNEMONIQUE	*OPERATION EFFECTUEE*	*Registre F*					
		S	*Z*	*H*	*P/V*	*N*	*C*
ADC HL,ss	Addition avec report de la paire de registres ss à HL.	-	-	X	V	0	-
ADC A,s	Addition avec report de l'opérande s à l'accumulateur.	-	-	-	V	0	-
ADD A,n	Addition de la valeur n à Acc.	-	-	-	V	0	-
ADD A,r	Addition du registre r à Acc.	-	-	-	V	0	-
ADD A,(HL)	Addition du contenu de l'adresse pointée par HL à Acc.	-	-	-	V	0	-
ADD A,(IX+d)	Addition du contenu de l'adresse pointée par IX plus déplacement d à Acc.	-	-	-	V	0	-
ADD A,(IY+d)	Addition du contenu de l'adresse pointée par IY plus déplacement d à Acc.	-	-	-	V	0	-
ADD HL,ss	Addition de la paire de reg. ss à HL.	●	●	X	●	0	-
ADD IX,pp	Addition de la paire de reg. pp à IX.	●	●	X	●	0	-
ADD IY,rr	Addition de la paire de reg. rr à IY.	●	●	X	●	0	-
AND s	ET logique entre l'opérande s et Acc.	-	-	1	P	0	0

Jeu d'instructions

MNEMONIQUE	OPERATION EFFECTUEE	Registre F					
		S	Z	H	P/V	N	C
BIT b,(HL)	Test du bit b du contenu de l'adresse pointée par HL.	X	-	1	X	0	•
BIT b,(IX+d)	Test du bit b du contenu de l'adresse pointée par IX plus déplacement d.	X	-	1	X	0	•
BIT b,(IY+d)	Test du bit b du contenu de l'adresse pointée par IY plus déplacement d.	X	-	1	X	0	•
BIT b,r	Test du bit b du reg. r.	X	-	1	X	0	•
CALL cc,nn	Appel d'un sous-programme à l'adresse nn si la condition cc est vraie.	•	•	•	•	•	•
CALL nn	Appel inconditionnel à un sous-programme à l'adresse nn.	•	•	•	•	•	•
CCF	Complémenter l'indicateur de report.	•	•	X	•	0	-
CP s	Comparer l'opérande s à Acc.	-	-	-	V	1	-
CPD	Comparer le contenu de l'adresse pointée par HL. Décrémenter HL et BC.	-	②	-	①	1	•
CPDR	Comparer le contenu de l'adresse pointée par HL. Décrémenter HL et BC, répéter jusqu'à ce que BC=0.	-	②	-	①	1	•
CPI	Comparer le contenu de l'adresse pointée par HL. Incrémenter HL et décrémenter BC.	-	②	-	①	1	0
CPIR	Comparer le contenu de l'adresse pointée par HL. Incrémenter HL et décrémenter BC, répéter jusqu'à ce que BC=0.	-	②	-	①	1	0
CPL	Complémenter Acc. (complément à 1).	•	•	1	•	1	•
DAA	Ajustement décimal de Acc.	-	-	-	P	•	-
DEC m	Décrémenter l'opérande m.	-	-	-	V	1	•
DEC IX	Décrémenter IX.	•	•	•	•	•	•
DEC IY	Décrémenter IY.	•	•	•	•	•	•
DEC ss	Décrémenter la paire de reg.ss.	•	•	•	•	•	•
DI	Désactive les interruptions.	•	•	•	•	•	•

Remarque *pour CPD, CPDR, CPI, CPIR*

1 → *le drapeau P/V=1 si le résultat de BC-1=0, autrement P/V=0*

2 → *le drapeau Z=1 si A=(HL), autrement Z=0.*

Jeu d'instructions

MNEMONIQUE	OPERATION EFFECTUEE	Registre F					
		S	Z	H	P/V	N	C
DJNZ e	Décrémenter B et effectuer un saut relatif si B≠0.	•	•	•	•	•	•
EI	Valider les interruptions.	•	•	•	•	•	•
EX (SP),HL	Echanger le sommet de la pile avec HL.	•	•	•	•	•	•
EX (SP),IX	Echange le sommet de la pile avec IX.	•	•	•	•	•	•
EX (SP),IY	Echange le sommet de la pile avec IY.	•	•	•	•	•	•
EX AF,AF'	Echange les contenus de AF et de AF'.	•	•	•	•	•	•
EX DE,HL	Echange les contenus de DE et de HL.	•	•	•	•	•	•
EXX	Echange les contenus de BC, DE HL avec les contenus de BC', DE', HL' respect.	•	•	•	•	•	•
HALT	Arrêt (attente d'une interruption ou d'une initialisation).	•	•	•	•	•	•
IM 0	Positionner le mode 0 des interruptions.	•	•	•	•	•	•
IM 1	Positionner le mode 1 des interruptions.	•	•	•	•	•	•
IM 2	Positionner le mode 2 des interruptions.	•	•	•	•	•	•
IN A,(n)	Charger Acc avec la donnée du port n.	•	•	•	•	•	•
IN r,(C)	Charger le reg. r avec la donnée du port adressé par C.	-	-	-	P	0	•
INC (HL)	Incrémenter le contenu de l'adresse pointée par HL.	-	-	-	V	0	•
INC IX	Incrémenter le reg. d'index IX.	•	•	•	•	•	•
INC (IX+d)	Incrémenter le contenu de l'adresse pointée par IX plus déplacement d.	-	-	-	V	0	•
INC IY	Incrémenter le reg. d'index IY.	•	•	•	•	•	•
INC (IY+d)	Incrémenter le contenu de l'adresse pointée par IY plus déplacement d.	-	-	-	V	0	•
INC r	Incrémenter reg. r.	-	-	-	V	0	•
INC ss	Incrémenter la paire de reg. ss.	•	•	•	•	•	•

MICROPROCESSEUR Z80

MNEMONIQUE	OPERATION EFFECTUEE	Registre F					
		S	Z	H	P/V	N	C
IND	Charger l'adresse pointée par HL avec la donnée provenant du port adressé par C. Décrémenter HL et B.	X	①	X	X	1	•
INDR	Charger l'adresse pointée par HL avec la donnée provenant du port adressé par C. Décrémenter HL et B et répéter jusqu'à ce que B=0.	X	1	X	X	1	•
INI	Charger l'adresse pointée par HL avec la donnée provenant du port adressé par C. Incrémenter HL et décrémenter B.	X	①	X	X	1	•
INIR	Charger l'adresse pointée par HL avec la donnée provenant du port adressé par C. Incrémenter HL et décrémenter B jusqu'à ce que B=0.	X	1	X	X	1	•
JP (HL)	Saut inconditionnel à l'adresse donnée par le contenu de HL.	•	•	•	•	•	•
JP (IX)	Saut inconditionnel à l'adresse indiquée par IX.	•	•	•	•	•	•
JP (IY)	Saut inconditionnel à l'adresse indiquée par IY.	•	•	•	•	•	•
JP cc,nn	Saut à l'adresse nn si la condition cc est vraie.	•	•	•	•	•	•
JP nn	Saut inconditionnel à l'adresse nn	•	•	•	•	•	•
JR C,e	Saut relatif à PC + e si c=1.	•	•	•	•	•	•
JR e	Saut inconditionnel relatif à PC + e.	•	•	•	•	•	•
JR NC,e	Saut relatif à PC + e si report = 0.	•	•	•	•	•	•
JR NZ,e	Saut relatif à PC + e si non zéro (Z=0).	•	•	•	•	•	•
JR Z,e	Saut relatif à PC + e si zéro (Z=1).	•	•	•	•	•	•
LD A,(BC)	Charger Acc avec le contenu de l'adresse pointée par la paire de reg. BC.	•	•	•	•	•	•
LD A,(DE)	Charger Acc avec le contenu de l'adresse pointée par la paire de reg. DE.	•	•	•	•	•	•

***Remarque** pour IND et INI*
① → *le drapeau Z=1 si le résultat de B-1=0, autrement Z=0*

MNEMONIQUE	OPERATION EFFECTUEE	Registre F					
		S	Z	H	P/V	N	C
LD A,I	Charger Acc avec I.	-	-	0	IFF*	0	•
LD A,(nn)	Charger Acc avec le contenu de l'adresse nn.	•	•	•	•	•	•
LD A,R	Charger Acc avec reg. R.	-	-	0	IFF*	0	•
LD (BC),A	Charger l'adresse pointée par BC avec Acc.	•	•	•	•	•	•
LD (DE),A	Charger l'adresse pointée par DE avec Acc.	•	•	•	•	•	•
LD (HL),n	Charger l'adresse pointée par HL avec la valeur n.	•	•	•	•	•	•
LD dd,nn	Charger la paire de reg. dd avec la valeur nn.	•	•	•	•	•	•
LD dd,(nn)	Charger la paire de reg. dd avec le contenu des adresses nn et nn+1.	•	•	•	•	•	•
LD HL,(nn)	Charger HL avec le contenu des adresses nn et nn+1.	•	•	•	•	•	•
LD (HL),r	Charger l'adresse pointée par HL avec le reg. r.	•	•	•	•	•	•
LD I,A	Charger le reg. I avec Acc.	•	•	•	•	•	•
LD IX,nn	Charger IX avec la valeur nn.	•	•	•	•	•	•
LD IX,(nn)	Charger IX avec le contenu des adresses nn et nn+1.	•	•	•	•	•	•
LD (IX+d),n	Charger l'adresse pointée par IX plus déplacement d avec la valeur n.	•	•	•	•	•	•
LD (IX+d),r	Charger l'adresse pointée par IX plus déplacement d avec le reg. r.	•	•	•	•	•	•
LD IY,nn	Charger IY avec la valeur nn.	•	•	•	•	•	•
LD IY,(nn)	Charger IY avec le contenu des adresses nn et nn+1.	•	•	•	•	•	•
LD (IY+d),n	Charger l'adresse pointée par IY plus déplacement d avec la valeur n.	•	•	•	•	•	•
LD (IY+d),r	Charger l'adresse pointée par IY plus déplacement avec le reg. r.	•	•	•	•	•	•
LD (nn),A	Charger l'adresse nn avec Acc.	•	•	•	•	•	•
LD (nn),dd	Charger les adresses nn et nn+1 avec la paire de reg. dd.	•	•	•	•	•	•
LD (nn),HL	Charger les adresses nn et nn+1 avec HL.	•	•	•	•	•	•

* *Le contenu de la bascule de validation des interruptions est copié dans le drapeau P/V.*

MNEMONIQUE	OPERATION EFFECTUEE	Registre F					
		S	Z	H	P/V	N	C
LD (nn),IX	Charger les adresses nn et nn+1 avec IX.	•	•	•	•	•	•
LD (nn),IY	Charger les adresses nn et nn+1 avec IY.	•	•	•	•	•	•
LD R,A	Charger le reg. R avec Acc.	•	•	•	•	•	•
LD r,(HL)	Charger le reg. r avec le contenu de l'adresse pointée par HL.	•	•	•	•	•	•
LD r,(IX+d)	Charger le reg. r avec le contenu de l'adresse pointée par IX plus déplacement d.	•	•	•	•	•	•
LD r,(IY+d)	Charger le reg. r avec le contenu de l'adresse pointée par IY plus déplacement d.	•	•	•	•	•	•
LD r,n	Charger le reg. r avec la valeur n.	•	•	•	•	•	•
LD r,r'	Charger le reg. r avec le reg. r'.	•	•	•	•	•	•
LD SP,HL	Charger le pointeur de pile avec HL.	•	•	•	•	•	•
LD SP,IX	Charger le pointeur de pile avec IX.	•	•	•	•	•	•
LD SP,IY	Charger le pointeur de pile avec IY.	•	•	•	•	•	•
LDD	Transférer le contenu de l'adresse pointée par DE dans l'adresse pointée par HL.	•	•	0	①	0	•
LDDR	Transférer le contenu de l'adresse pointée par DE dans l'adresse pointée par HL. Décrémenter DE,HL et BC ; répéter jusqu'à ce que BC=0.	•	•	0	0	0	•
LDI	Transférer le contenu de l'adresse pointée par DE dans l'adresse pointée par HL. Incrémenter HL,DE et décrémenter BC.	•	•	0	①	0	•
LDIR	Transférer le contenu de l'adresse pointée par DE dans l'adresse pointée par HL. Incrémenter DE,HL et décrémenter BC ; répéter jusqu'à ce que BC=0.	•	•	0	0	0	•

Remarque *pour LDD et LDI*

① → *le drapeau P/V=0 si le résultat de BC-1=0, autrement P/V=1*

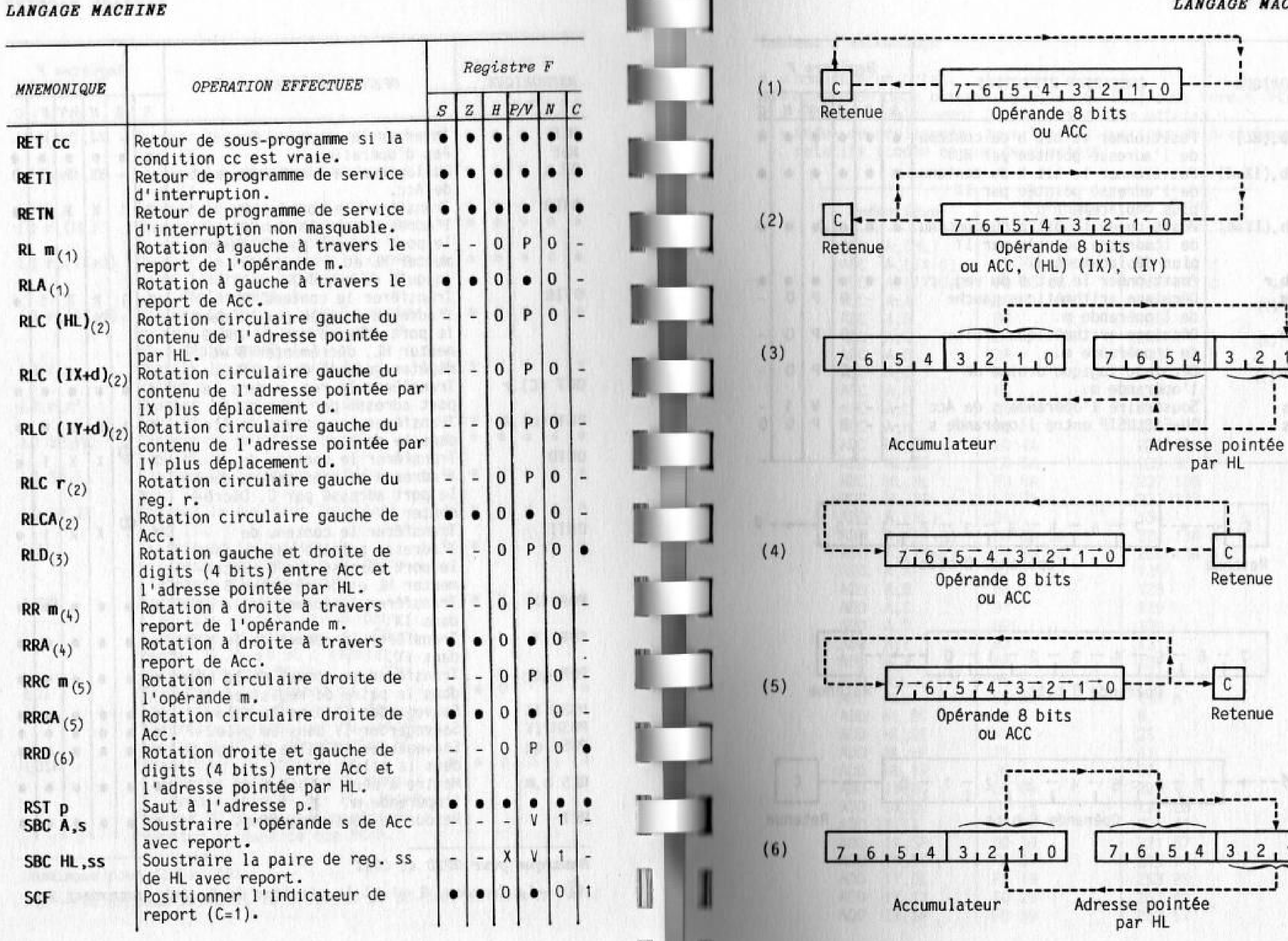
MNEMONIQUE	OPERATION EFFECTUEE	Registre F					
		S	Z	H	P/V	N	C
NEG	Inverser le signe de Acc.	-	-	-	V	1	-
NOP	Pas d'opération.	•	•	•	•	•	•
OR s	OU logique de l'opérande s et de Acc.	-	-	0	P	0	0
OTDR	Transférer le contenu de l'adresse pointée par HL dans le port adressé par C. Décrémenter HL et B ; répéter jusqu'à ce que B=0.	X	1	X	X	1	•
OTIR	Transférer le contenu de l'adresse pointée par HL dans le port adressé par C. Incrémenter HL, décrémenter B et répéter jusqu'à ce que B=0.	X	1	X	X	1	•
OUT (C),r	Transférer le reg. r dans le port adressé par le reg. C.	•	•	•	•	•	•
OUT (n),A	Transférer le contenu de Acc dans le port n.	•	•	•	•	•	•
OUTD	Transférer le contenu de l'adresse pointée par HL dans le port adressé par C. Décrémenter HL et B.	X	②	X	X	1	•
OUTI	Transférer le contenu de l'adresse pointée par HL dans le port adressé par C. Incrémenter HL et décrémenter B.	X	②	X	X	1	•
POP IX	Transférer le sommet de la pile dans IX.	•	•	•	•	•	•
POP IY	Transférer le sommet de la pile dans IY.	•	•	•	•	•	•
POP qq	Transférer le sommet de la pile dans la paire de registres qq.	•	•	•	•	•	•
PUSH IX	Sauvegarder IX dans la pile.	•	•	•	•	•	•
PUSH IY	Sauvegarder IY dans la pile.	•	•	•	•	•	•
PUSH qq	Sauvegarder la paire de reg. qq dans la pile.	•	•	•	•	•	•
RES b,m	Mettre à zéro le bit b de l'opérande m.	•	•	•	•	•	•
RET	Retour de sous-programme.	•	•	•	•	•	•

Remarque *pour OUTD et OUTI*

② →*le drapeau Z =1 si le résultat de B 1=0, autrement Z=0*

MICROPROCESSEUR Z80

MNEMONIQUE	OPERATION EFFECTUEE	Registre F					
		S	Z	H	P/V	N	C
RET cc	Retour de sous-programme si la condition cc est vraie.	●	●	●	●	●	●
RETI	Retour de programme de service d'interruption.	●	●	●	●	●	●
RETN	Retour de programme de service d'interruption non masquable.	●	●	●	●	●	●
RL m$_{(1)}$	Rotation à gauche à travers le report de l'opérande m.	-	-	0	P	0	-
RLA$_{(1)}$	Rotation à gauche à travers le report de Acc.	●	●	0	●	0	-
RLC (HL)$_{(2)}$	Rotation circulaire gauche du contenu de l'adresse pointée par HL.	-	-	0	P	0	-
RLC (IX+d)$_{(2)}$	Rotation circulaire gauche du contenu de l'adresse pointée par IX plus déplacement d.	-	-	0	P	0	-
RLC (IY+d)$_{(2)}$	Rotation circulaire gauche du contenu de l'adresse pointée par IY plus déplacement d.	-	-	0	P	0	-
RLC r$_{(2)}$	Rotation circulaire gauche du reg. r.	-	-	0	P	0	-
RLCA$_{(2)}$	Rotation circulaire gauche de Acc.	●	●	0	●	0	-
RLD$_{(3)}$	Rotation gauche et droite de digits (4 bits) entre Acc et l'adresse pointée par HL.	-	-	0	P	0	●
RR m$_{(4)}$	Rotation à droite à travers report de l'opérande m.	-	-	0	P	0	-
RRA$_{(4)}$	Rotation à droite à travers report de Acc.	●	●	0	●	0	-
RRC m$_{(5)}$	Rotation circulaire droite de l'opérande m.	-	-	0	P	0	-
RRCA$_{(5)}$	Rotation circulaire droite de Acc.	●	●	0	●	0	-
RRD$_{(6)}$	Rotation droite et gauche de digits (4 bits) entre Acc et l'adresse pointée par HL.	-	-	0	P	0	●
RST p	Saut à l'adresse p.	●	●	●	●	●	●
SBC A,s	Soustraire l'opérande s de Acc avec report.	-	-	-	V	1	-
SBC HL,ss	Soustraire la paire de reg. ss de HL avec report.	-	-	X	V	1	-
SCF	Positionner l'indicateur de report (C=1).	●	●	0	●	0	1



MNEMONIQUE	OPERATION EFFECTUEE	Registre F					
		S	Z	H	P/V	N	C
SET b,(HL)	Positionner le bit b du contenu de l'adresse pointée par HL.	•	•	•	•	•	•
SET b,(IX+d)	Positionner le bit b du contenu de l'adresse pointée par IX plus déplacement d.	•	•	•	•	•	•
SET b,(IY+d)	Positionner le bit b du contenu de l'adresse pointée par IY plus déplacement d.	•	•	•	•	•	•
SET b,r	Positionner le bit b du reg. r.	•	•	•	•	•	•
SLA m (7)	Décalage arithmétique gauche de l'opérande m.	-	-	0	P	0	-
SRA m (8)	Décalage arithmétique droite de l'opérande m.	-	-	0	P	0	-
SRL m (9)	Décalage logique droite de l'opérande m.	-	-	0	P	0	-
SUB s	Soustraire l'opérande s de Acc	-	-	-	V	1	-
XOR s	OU-EXCLUSIF entre l'opérande s et Acc.	-	-	0	P	0	0

(7) C ←--- 7 ← 6 ← 5 ← 4 ← 3 ← 2 ← 1 ← 0 ←--- Ø

Retenue Opérande 8 bits

(8) 7 → 6 → 5 → 4 → 3 → 2 → 1 → 0 ---→ C

Opérande 8 bits Retenue

(9) Ø ---→ 7 → 6 → 5 → 4 → 3 → 2 → 1 → 0 ---→ C

Opérande 8 bits Retenue

Tableau d'assemblage

n = nombre d'un octet.
nn = nombre de deux octets (adresse 16 bits) sous format OMS,OPS.
d = index de déplacement (codé dd sur deux octets).
e = index de déplacement pour les instructions de branchements relatifs (codée **ee** sur deux octets).

Mnémonique	Code hexadécimal	Code décimal
ADC A,(HL)	8E	142
ADC A,(IX+d)	DD 8E dd	221 142 d
ADC A,(IY+d)	FD 8E dd	253 142 d
ADC A,A	8F	143
ADC A,B	88	136
ADC A,C	8A	137
ADC A,D	9A	138
ADC A,E	8B	139
ADC A,H	8C	140
ADC A,L	8D	141
ADC A,n	CE nn	206 n
ADC HL,BC	ED 4A	237 74
ADC HL,DE	ED 5A	237 90
ADC HL,HL	ED 6A	237 106
ADC HL,SP	ED 7A	237 122
ADD A,(HL)	86	134
ADD A,(IX+d)	DD 86 dd	221 134 d
ADD A,(IY+d)	FD 86 dd	253 134 d
ADD A,A	87	135
ADD A,B	80	128
ADD A,C	81	129
ADD A,D	82	130
ADD A,E	83	131
ADD A,H	84	132
ADD A,L	85	133
ADD A,n	C6 nn	198 n
ADD HL,BC	09	9
ADD HL,DE	19	25
ADD HL,HL	29	41
ADD HL,SP	39	57
ADD IX,BC	DD 09	221 9
ADD IX,DE	DD 19	221 25
ADD IX,IX	DD 29	221 41
ADD IX,SP	DD 39	221 57
ADD IY,BC	FD 09	253 9
ADD IY,DE	FD 19	253 25
ADD IY,IY	FD 29	253 41
ADD IY,SP	FD 39	253 57

MICROPROCESSEUR Z80

Mnémonique	Code hexadécimal	Code décimal
AND (HL)	A6	166
AND (IX+d)	DD A6 dd	221 166 d
AND (IY+d)	FD A6 dd	253 166 d
AND A	A7	167
AND B	A0	160
AND C	A1	161
AND D	A2	162
AND E	A3	163
AND H	A4	164
AND L	A5	165
AND n	E6 nn	230 n
BIT 0,(HL)	CB 46	203 70
BIT 0,(IX+d)	DD CB dd 46	221 203 d 70
BIT 0,(IY+d)	FD CB dd 46	253 203 d 70
BIT 0,A	CB 47	203 71
BIT 0,B	CB 40	203 64
BIT 0,C	CB 41	203 65
BIT 0,D	CB 42	203 66
BIT 0,E	CB 43	203 67
BIT 0,H	CB 44	203 68
BIT 0,L	CB 45	203 69
BIT 1,(HL)	CB 4E	203 78
BIT 1,(IX+d)	DD CB dd 4E	221 203 d 78
BIT 1,(IY+d)	FD CB dd 4E	253 203 d 78
BIT 1,A	CB 4F	203 79
BIT 1,B	CB 48	203 72
BIT 1,C	CB 49	203 73
BIT 1,D	CB 4A	203 74
BIT 1,E	CB 4B	203 75
BIT 1,H	CB 4C	203 76
BIT 1,L	CB 4D	203 77
BIT 2,(HL)	CB 56	203 86
BIT 2,(IX+d)	DD CB dd 56	221 203 d 86
BIT 2,(IY+d)	FD CB dd 56	253 203 d 86
BIT 2,A	CB 57	203 87
BIT 2,B	CB 50	203 80
BIT 2,C	CB 51	203 81
BIT 2,D	CB 52	203 82
BIT 2,E	CB 53	203 83
BIT 2,H	CB 54	203 84
BIT 2,L	CB 55	203 85
BIT 3,(HL)	CB 5E	203 94
BIT 3,(IX+d)	DD CB dd 5E	221 203 d 94
BIT 3,(IY+d)	FD CB dd 5E	253 203 d 94
BIT 3,A	CB 5F	203 95
BIT 3,B	CB 58	203 88
BIT 3,C	CB 59	203 89

Mnémonique	Code hexadécimal	Code décimal
BIT 3,D	CB 5A	203 90
BIT 3,E	CB 5B	203 91
BIT 3,H	CB 5C	203 92
BIT 3,L	CB 5D	203 93
BIT 4,(HL)	CB 66	203 102
BIT 4,(IX+d)	DD CB dd 66	221 203 d 102
BIT 4,(IY+d)	FD CB dd 86	253 203 d 102
BIT 4,A	CB 67	203 103
BIT 4,B	CB 60	203 96
BIT 4,C	CB 61	203 97
BIT 4,D	CB 62	203 98
BIT 4,E	CB 63	203 99
BIT 4,H	CB 64	203 100
BIT 4,L	CB 65	203 101
BIT 5,(HL)	CB 6E	203 110
BIT 5,(IX+d)	DD CB dd 6E	221 203 d 110
BIT 5,(IY+d)	FD CB dd 6E	253 203 d 110
BIT 5,A	CB 6F	203 111
BIT 5,B	CB 68	203 104
BIT 5,C	CB 69	203 105
BIT 5,D	CB 6A	203 106
BIT 5,E	CB 6B	203 107
BIT 5,H	CB 6C	203 108
BIT 5,L	CB 6D	203 109
BIT 6,(HL)	CB 76	203 118
BIT 6,(IX+d)	DD CB dd 76	221 203 d 118
BIT 6,(IY+d)	FD CB dd 76	253 203 d 118
BIT 6,A	CB 77	203 119
BIT 6,B	CB 70	203 112
BIT 6,C	CB 71	203 113
BIT 6,D	CB 72	203 114
BIT 6,E	CB 73	203 115
BIT 6,H	CB 74	203 116
BIT 6,L	CB 75	203 117
BIT 7,(HL)	CB 7E	203 126
BIT 7,(IX+d)	DD CB dd 7E	221 203 d 126
BIT 7,(IY+d)	FD CB dd 7E	253 203 d 126
BIT 7,A	CB 7F	203 127
BIT 7,B	CB 78	203 120
BIT 7,C	CB 79	203 121
BIT 7,D	CB 7A	203 122
BIT 7,E	CB 7B	203 123
BIT 7,H	CB 7C	203 124
BIT 7,L	CB 7D	203 125
CALL C,nn	DC nn nn	220 n n
CALL M,nn	FC nn nn	252 n n

Mnémonique	Code hexadécimal	Code décimal
CALL NC,nn	D4 nn nn	212 n n
CALL nn	CD nn nn	205 n n
CALL NZ,nn	C4 nn nn	196 n n
CALL P,nn	F4 nn nn	244 n n
CALL PE,nn	EC nn nn	236 n n
CALL PO,nn	E4 nn nn	228 n n
CALL Z,nn	CC nn nn	204 n n
CCF	3F	63
CP (HL)	BE	190
CP (IX+d)	DD BE dd	221 190 d
CP (IY+d)	FD BE dd	253 190 d
CP A	BF	191
CP B	B8	184
CP C	B9	185
CP D	BA	186
CP E	BB	187
CP H	BC	188
CP L	BD	189
CP n	FE nn	254 n
CPD	ED A9	237 169
CPDR	ED B9	237 185
CPI	ED A1	237 161
CPIR	ED B1	237 177
CPL	2F	47
DAA	27	39
DEC (HL)	35	53
DEC (IX+d)	DD 35 dd	221 53 d
DEC (IY+d)	FD 35 dd	253 53 d
DEC A	3D	61
DEC B	05	5
DEC BC	0B	11
DEC C	0D	13
DEC D	15	21
DEC DE	1B	27
DEC E	1D	29
DEC H	25	37
DEC HL	2B	43
DEC IX	DD 2B	221 43
DEC IY	FD 2B	253 43
DEC L	2D	45
DEC SP	3B	59
DI	F3	243
DJNZ e	10 ee	16 e
EI	FB	251
EX (SP),HL	E3	227
EX (SP),IX	DD E3	221 227

Mnémonique	Code hexadécimal	Code décimal
EX (SP),IY	FD E3	253 227
EX AF,AF'	08	8
EX DE,HL	EB	235
EXX	D9	217
HALT	76	118
IM 0	ED 46	237 70
IM 1	ED 56	237 86
IM 2	ED 5E	237 94
IN A,(C)	ED 78	237 120
IN A,n	DB nn	219 n
IN B,(C)	ED 40	237 64
IN C,(C)	ED 48	237 72
IN D,(C)	ED 50	237 80
IN E,(C)	ED 58	237 88
IN H,(C)	ED 60	237 96
IN L,(C)	ED 68	237 104
INC (HL)	34	52
INC (IX+d)	DD 34 dd	221 52 d
INC (IY+d)	FD 34 dd	253 52 d
INC A	3C	60
INC B	04	4
INC BC	03	3
INC C	0C	12
INC D	14	20
INC DE	13	19
INC E	1C	28
INC H	24	36
INC HL	23	35
INC IX	DD 23	221 35
INC IY	FD 23	253 35
INC L	2C	44
INC SP	33	51
IND	ED AA	237 170
INDR	ED BA	237 186
INI	ED A2	237 162
INIR	ED B2	237 178
JP (HL)	E9	233
JP (IX)	DD E9	221 233
JP (IY)	FD E9	253 233
JP C,nn	DA nn nn	218 n n
JP M,nn	FA nn nn	250 n n
JP NC,nn	D2 nn nn	210 n n
JP nn	C3 nn nn	195 n n
JP NZ,nn	C2 nn nn	194 n n
JP P,nn	F2 nn nn	242 n n
JP PE,nn	EA nn nn	234 n n

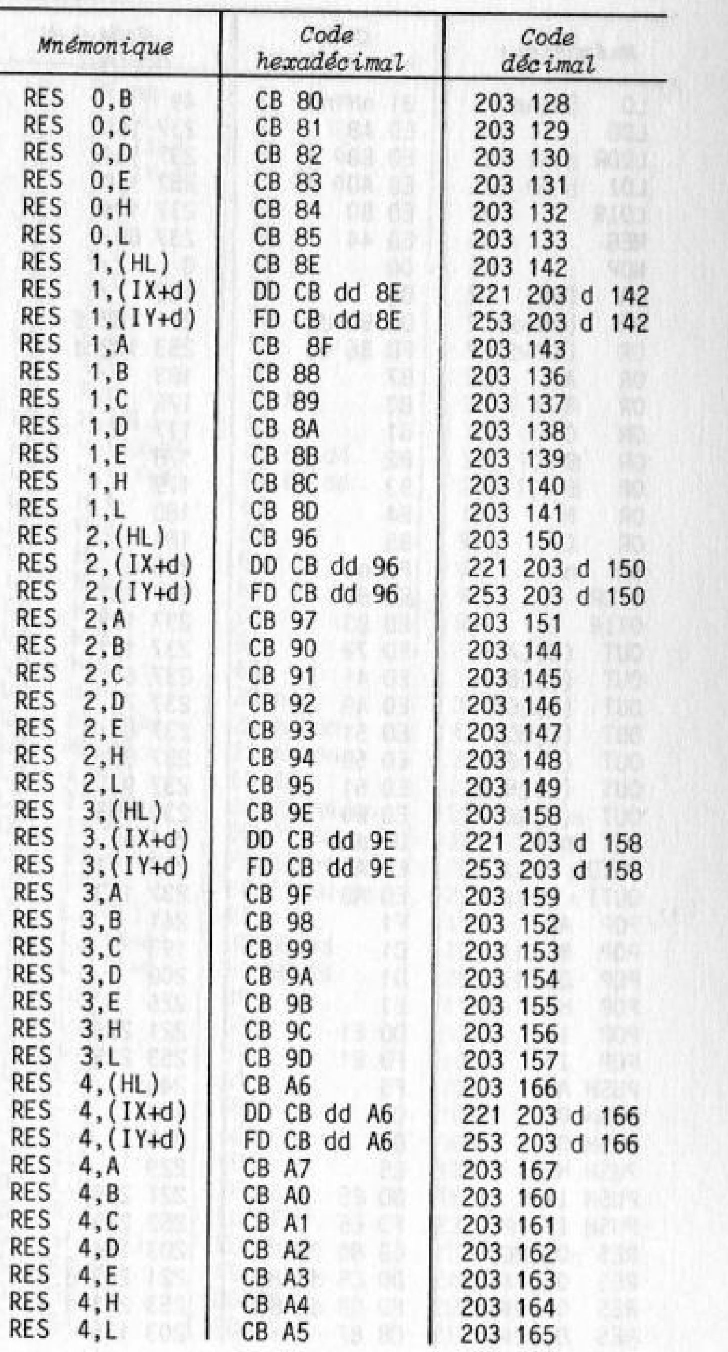
Mnémonique	Code hexadécimal	Code décimal
JP PO,nn	E2 nn nn	226 n n
JP Z,nn	CA nn nn	202 n n
JR C,e	38 ee	56 e
JR e	18 ee	24 e
JR NC,e	30 ee	48 e
JR NZ,e	20 ee	32 e
JR Z,e	28 ee	40 e
LD (BC),A	02	2
LD (DE),A	12	18
LD (HL),A	77	119
LD (HL),B	70	112
LD (HL),C	71	113
LD (HL),D	72	114
LD (HL),E	73	115
LD (HL),H	74	116
LD (HL),L	75	117
LD (HL),n	36 nn	54 n
LD (IX+d),A	DD 77 dd	221 119 d
LD (IX+d),B	DD 70 dd	221 112 d
LD (IX+d),C	DD 71 dd	221 113 d
LD (IX+d),D	DD 72 dd	221 114 d
LD (IX+d),E	DD 73 dd	221 115 d
LD (IX+d),H	DD 74 dd	221 116 d
LD (IX+d),L	DD 75 dd	221 117 d
LD (IX+d),n	DD 36 dd nn	221 54 d n
LD (IY+d),A	FD 77 dd	253 119 d
LD (IY+d),B	FD 70 dd	253 112 d
LD (IY+d),C	FD 71 dd	253 113 d
LD (IY+d),D	FD 72 dd	253 114 d
LD (IY+d),E	FD 73 dd	253 115 d
LD (IY+d),H	FD 74 dd	253 116 d
LD (IY+d),L	FD 75 dd	253 117 d
LD (IY+d),n	FD 36 dd nn	253 54 d n
LD (nn),A	32 nn nn	50 n n
LD (nn),BC	ED 43 nn nn	237 67 n n
LD (nn),DE	ED 53 nn nn	237 83 n n
LD (nn),HL	22 nn nn	34 n n
LD (nn),IX	DD 22 nn nn	221 34 n n
LD (nn),IY	FD 22 nn nn	253 34 n n
LD (nn),SP	ED 73 nn nn	237 115 n n
LD A,(BC)	0A	10
LD A,(DE)	1A	26
LD A,(HL)	7E	126
LD A,(IX+d)	DD 7E dd	221 126 d
LD A,(IY+d)	FD 7E dd	253 126 d
LD A,(nn)	3A nn nn	58

Mnémonique	Code hexadécimal	Code décimal
LD A,A	7F	127
LD A,B	78	120
LD A,C	79	121
LD A,D	7A	122
LD A,E	7B	123
LD A,H	7C	124
LD A,I	ED 57	237 87
LD A,L	7D	125
LD A,n	3E nn	62 n
LD A,R	ED 5F	237 95
LD B,(HL)	46	70
LD B,(IX+d)	DD 46 dd	221 70 d
LD B,(IY+d)	FD 46 dd	253 70 d
LD B,A	47	71
LD B,B	40	64
LD B,C	41	65
LD B,D	42	66
LD B,E	43	67
LD B,H	44	68
LD B,L	45	69
LD B,n	06 nn	6 n
LD BC,(nn)	ED 4B nn nn	237 75 n n
LD BC,nn	01 nn nn	1 n n
LD C,(HL)	4E	78
LD C,(IX+d)	DD 4E dd	221 78 d
LD C,(IY+d)	FD 4E dd	253 78 d
LD C,A	4F	79
LD C,B	48	72
LD C,C	49	73
LD C,D	4A	74
LD C,E	4B	75
LD C,H	4C	76
LD C,L	4D	77
LD C,n	0E nn	14 n
LD D,(HL)	56	86
LD D,(IX+d)	DD 56 dd	221 86 d
LD D,(IY+d)	FD 56 dd	253 86 d
LD D,A	57	87
LD D,B	50	80
LD D,C	51	81
LD D,D	52	82
LD D,E	53	83
LD D,H	54	84
LD D,L	55	85
LD D,n	16 nn	22
LD DE,(nn)	ED 5B nn nn	237 91 n n

MICROPROCESSEUR Z80

Mnémonique	Code hexadécimal	Code décimal
LD DE,nn	11 nn nn	17 n n
LD E,(HL)	5E	94
LD E,(IX+d)	DD 5E dd	221 94 d
LD E,(IY+d)	FD 5E dd	253 94 d
LD E,A	5F	95
LD E,B	58	88
LD E,C	59	89
LD E,D	5A	90
LD E,E	5B	91
LD E,H	5C	92
LD E,L	5D	93
LD E,n	1E nn	30 n n
LD H,(HL)	66	102
LD H,(IX+d)	DD 66 dd	221 102 d
LD H,(IY+d)	FD 66 dd	253 102 d
LD H,A	67	103
LD H,B	60	96
LD H,C	61	97
LD H,D	62	98
LD H,E	63	99
LD H,H	64	100
LD H,L	65	101
LD H,n	26 nn	38 n
LD HL,(nn)	2A nn nn	42 n n
LD HL,nn	21 nn nn	33 n n
LD I,A	ED 47	237 71
LD IX,(nn)	DD 2A nn nn	221 42 n n
LD IX,nn	DD 21 nn nn	221 33 n n
LD IY,(nn)	FD 2A nn nn	253 42 n n
LD IY,nn	FD 21 nn nn	253 33 n n
LD L,(HL)	6E	110
LD L,(IX+d)	DD 6E dd	221 110 d
LD L,(IY+d)	FD 6E dd	253 110 d
LD L,A	6F	111
LD L,B	68	104
LD L,C	69	105
LD L,D	6A	106
LD L,E	6B	107
LD L,H	6C	108
LD L,L	6D	109
LD L,n	2E nn	46 n
LD R,A	ED 4F	237 79
LD SP,(nn)	ED 7B nn nn	237 123 n n
LD SP,HL	F9	249
LD SP,IX	DD F9	221 249
LD SP,IY	FD F9	253 249

Mnémonique	Code hexadécimal	Code décimal
LD SP,nn	31 nn nn	49 n n
LDD	ED A8	237 168
LDDR	ED B8	237 184
LDI	ED A0	237 160
LDIR	ED B0	237 176
NEG	ED 44	237 68
NOP	00	0
OR (HL)	B6	182
OR (IX+d)	DD B6 dd	221 182 d
OR (IY+d)	FD B6 dd	253 182 d
OR A	B7	183
OR B	B0	176
OR C	B1	177
OR D	B2	178
OR E	B3	179
OR H	B4	180
OR L	B5	181
OR n	F6 nn	246 n
OTDR	ED BB	237 187
OTIR	ED B3	237 179
OUT (C),A	ED 79	237 121
OUT (C),B	ED 41	237 65
OUT (C),C	ED 49	237 73
OUT (C),D	ED 51	237 81
OUT (C),E	ED 59	237 89
OUT (C),H	ED 61	237 97
OUT (C),L	ED 69	237 105
OUT n,A	D3 nn	211 n
OUTD	ED AB	237 171
OUTI	ED A3	237 163
POP AF	F1	241
POP BC	C1	193
POP DE	D1	209
POP HL	E1	225
POP IX	DD E1	221 225
POP IY	FD E1	253 225
PUSH AF	F5	245
PUSH BC	C5	197
PUSH DE	D5	213
PUSH HL	E5	229
PUSH IX	DD E5	221 229
PUSH IY	FD E5	253 229
RES 0,(HL)	CB 86	203 134
RES 0,(IX+d)	DD CB dd 86	221 203 d 134
RES 0,(IY+d)	FD CB dd 86	253 203 d 134
RES 0,A	CB 87	203 135



Mnémonique	Code hexadécimal	Code décimal
RES 0,B	CB 80	203 128
RES 0,C	CB 81	203 129
RES 0,D	CB 82	203 130
RES 0,E	CB 83	203 131
RES 0,H	CB 84	203 132
RES 0,L	CB 85	203 133
RES 1,(HL)	CB 8E	203 142
RES 1,(IX+d)	DD CB dd 8E	221 203 d 142
RES 1,(IY+d)	FD CB dd 8E	253 203 d 142
RES 1,A	CB 8F	203 143
RES 1,B	CB 88	203 136
RES 1,C	CB 89	203 137
RES 1,D	CB 8A	203 138
RES 1,E	CB 8B	203 139
RES 1,H	CB 8C	203 140
RES 1,L	CB 8D	203 141
RES 2,(HL)	CB 96	203 150
RES 2,(IX+d)	DD CB dd 96	221 203 d 150
RES 2,(IY+d)	FD CB dd 96	253 203 d 150
RES 2,A	CB 97	203 151
RES 2,B	CB 90	203 144
RES 2,C	CB 91	203 145
RES 2,D	CB 92	203 146
RES 2,E	CB 93	203 147
RES 2,H	CB 94	203 148
RES 2,L	CB 95	203 149
RES 3,(HL)	CB 9E	203 158
RES 3,(IX+d)	DD CB dd 9E	221 203 d 158
RES 3,(IY+d)	FD CB dd 9E	253 203 d 158
RES 3,A	CB 9F	203 159
RES 3,B	CB 98	203 152
RES 3,C	CB 99	203 153
RES 3,D	CB 9A	203 154
RES 3,E	CB 9B	203 155
RES 3,H	CB 9C	203 156
RES 3,L	CB 9D	203 157
RES 4,(HL)	CB A6	203 166
RES 4,(IX+d)	DD CB dd A6	221 203 d 166
RES 4,(IY+d)	FD CB dd A6	253 203 d 166
RES 4,A	CB A7	203 167
RES 4,B	CB A0	203 160
RES 4,C	CB A1	203 161
RES 4,D	CB A2	203 162
RES 4,E	CB A3	203 163
RES 4,H	CB A4	203 164
RES 4,L	CB A5	203 165

Mnémonique	Code hexadécimal	Code décimal
RES 5,(HL)	CB AE	203 174
RES 5,(IX+d)	DD CB dd AE	221 203 d 174
RES 5,(IY+d)	FD CB dd AE	253 203 d 174
RES 5,A	CB AF	203 175
RES 5,B	CB A8	203 168
RES 5,C	CB A9	203 169
RES 5,D	CB AA	203 170
RES 5,E	CB AB	203 171
RES 5,H	CB AC	203 172
RES 5,L	CB AD	203 173
RES 6,(HL)	CB B6	203 182
RES 6,(IX+d)	DD CB dd B6	221 203 d 182
RES 6,(IY+d)	FD CB dd B6	253 203 d 182
RES 6,A	CB B7	203 183
RES 6,B	CB B0	203 176
RES 6,C	CB B1	203 177
RES 6,D	CB B2	203 178
RES 6,E	CB B3	203 179
RES 6,H	CB B4	203 180
RES 6,L	CB B5	203 181
RES 7,(HL)	CB BE	203 190
RES 7,(IX+d)	DD CB dd BE	221 203 d 190
RES 7,(IY+d)	FD CB dd BE	253 203 d 190
RES 7,A	CB BF	203 191
RES 7,B	CB B8	203 184
RES 7,C	CB B9	203 185
RES 7,D	CB BA	203 186
RES 7,E	CB BB	203 187
RES 7,H	CB BC	203 188
RES 7,L	CB BD	203 189
RET	C9	201
RET C	D8	216
RET M	F8	248
RET NC	D0	208
RET NZ	C0	192
RET P	F0	240
RET PE	E8	232
RET PO	E0	224
RET Z	C8	200
RETI	ED 4D	237 77
RETN	ED 45	237 69
RL (HL)	CB 16	203 22
RL (IX+d)	DD CB dd 16	221 203 d 22
RL (IY+d)	FD CB dd 16	253 203 d 22
RL A	CB 17	203 23
RL B	CB 10	203 16

Mnémonique	Code hexadécimal	Code décimal
RL C	CB 11	203 17
RL D	CB 12	203 18
RL E	CB 13	203 19
RL L	CB 15	203 20
RLA	17	23
RLC (HL)	CB 06	203 6
RLC (IX+d)	DD CB dd 06	221 203 d 6
RLC (IY+d)	FD CB dd 06	253 203 d 6
RLC A	CB 07	203 7
RLC B	CB 00	203 0
RLC C	CB 01	203 1
RLC D	CB 02	203 2
RLC E	CB 03	203 3
RLC H	CB 04	203 4
RLC L	CB 05	203 5
RLCA	07	7
RLD	ED 6F	237 111
RR (HL)	CB 1E	203 30
RR (IX+d)	DD CB dd 1E	221 203 d 30
RR (IY+d)	FD CB dd 1E	253 203 d 30
RR A	CB 1F	203 31
RR B	CB 18	203 24
RR C	CB 19	203 25
RR D	CB 1A	203 26
RR E	CB 1B	203 27
RR H	CB 1C	203 28
RR L	CB 1D	203 29
RRA	1F	31
RRC (HL)	CB 0E	203 14
RRC (IX+d)	DD CB dd 0E	221 203 d 14
RRC (IY+d)	FD CB dd 0E	253 203 d 14
RRC A	CB 0F	203 15
RRC B	CB 08	203 8
RRC C	CB 09	203 9
RRC D	CB 0A	203 10
RRC E	CB 0B	203 11
RRC H	CB 0C	203 12
RRC L	CB 0D	203 13
RRCA	0F	15
RRD	ED 67	237 103
RST 00H	C7	199
RST 08H	CF	207
RST 10H	D7	215
RST 18H	DF	223
RST 20H	E7	231
RST 28H	EF	239

Mnémonique	Code hexadécimal	Code décimal
RST 30 H	F7	247
RST 38H	FF	255
SBC A,(HL)	9E	158
SBC A,(IX+d)	DD 9E dd	221 158 d
SBC A,(IY+d)	FD 9E dd	253 158 d
SBC A,A	9F	159
SBC A,B	98	152
SBC A,C	99	153
SBC A,D	9A	154
SBC A,E	9B	155
SBC A,H	9C	156
SBC A,L	9D	157
SBC A,n	DE nn	222 n
SBC HL,BC	ED 42	237 66
SBC HL,DE	ED 52	237 82
SBC HL,HL	ED 62	237 98
SBC HL,SP	ED 72	237 114
SCF	37	55
SET 0,(HL)	CB C6	203 198
SET 0,(IX+d)	DD CB dd C6	221 203 d 198
SET 0,(IY+d)	FD CB dd C6	253 203 d 198
SET 0,A	CB C7	203 199
SET 0,B	CB C0	203 192
SET 0,C	CB C1	203 193
SET 0,D	CB C2	203 194
SET 0,E	CB C3	203 195
SET 0,H	CB C4	203 196
SET 0,L	CB C5	203 197
SET 1,(HL)	CB CE	203 206
SET 1,(IX+d)	DD CB dd CE	221 203 d 206
SET 1,(IY+d)	FD CB dd CE	253 203 d 206
SET 1,A	CB CF	203 207
SET 1,B	CB C8	203 200
SET 1,C	CB C9	203 201
SET 1,D	CB CA	203 202
SET 1,E	CB CB	203 203
SET 1,H	CB CC	203 204
SET 1,L	CB CD	203 205
SET 2,(HL)	CB D6	203 214
SET 2,(IX+d)	DD CB dd D6	221 203 d 214
SET 2,(IY+d)	FD CB dd D6	253 203 d 214
SET 2,A	CB D7	203 215
SET 2,B	CB D0	203 208
SET 2,C	CB D1	203 209
SET 2,D	CB D2	203 210
SET 2,E	CB D3	203 211

Mnémonique	Code hexadécimal	Code décimal
SET 2,H	CB D4	203 212
SET 2,L	CB D5	203 213
SET 3,(HL)	CB DE	203 222
SET 3,(IX+d)	DD CB dd DE	221 203 d 222
SET 3,(IY+d)	FD CB dd DE	253 203 d 222
SET 3,A	CB DF	203 223
SET 3,B	CB D8	203 216
SET 3,C	CB D9	203 217
SET 3,D	CB DA	203 218
SET 3,E	CB DB	203 219
SET 3,H	CB DC	203 220
SET 3,L	CB DD	203 221
SET 4,(HL)	CB E6	203 230
SET 4,(IX+d)	DD CB dd E6	221 203 d 230
SET 4,(IY+d)	FD CB dd E6	253 203 d 230
SET 4,A	CB E7	203 231
SET 4,B	CB E0	203 224
SET 4,C	CB E1	203 225
SET 4,D	CB E2	203 226
SET 4,C	CB E3	203 227
SET 4,H	CB E4	203 228
SET 4,L	CB E5	203 229
SET 5,(HL)	CB EE	203 238
SET 5,(IX+d)	DD CB dd EE	221 203 d 238
SET 5,(IY+d)	FD CB dd 238	253 203 d 238
SET 5,A	CB EF	203 239
SET 5,B	CB E8	203 232
SET 5,C	CB E9	203 233
SET 5,D	CB EA	203 234
SET 5,E	CB EB	203 235
SET 5,H	CB EC	203 236
SET 5,L	CB ED	203 237
SET 6,(HL)	CB F6	203 246
SET 6,(IX+d)	DD CD dd F6	221 203 d 246
SET 6,(IY+d)	FD CB dd F6	253 203 d 246
SET 6,A	CB F7	203 247
SET 6,B	CB F0	203 240
SET 6,C	CB F1	203 241
SET 6,D	CB F2	203 242
SET 6,E	CB F3	203 243
SET 6,H	CB F4	203 244
SET 6,L	CB F5	203 245
SET 7,(HL)	CB FE	203 254
SET 7,(IX+d)	DD CB dd FE	221 203 d 254
SET 7,(IY+d)	FD CB dd FE	253 203 d 254
SET 7,A	CB FF	203 255

Mnémonique	Code hexadécimal	Code décimal
SET 7,B	CB F8	203 248
SET 7,C	CB F9	203 249
SET 7,D	CB FA	203 250
SET 7,E	CB FB	203 251
SET 7,H	CB FC	203 252
SET 7,L	CB FD	203 253
SLA (HL)	CB 26	203 38
SLA (IX+d)	DD CB dd 26	221 203 d 38
SLA (IY+d)	FD CB dd 26	253 203 d 38
SLA A	CB 27	203 39
SLA B	CB 20	203 32
SLA C	CB 21	203 33
SLA D	CB 22	203 34
SLA E	CB 23	203 35
SLA H	CB 24	203 36
SLA L	CB 25	203 37
SRA (HL)	CB 2E	203 46
SRA (IX+d)	DD CB dd 2E	221 203 d 46
SRA (IY+d)	FD CB dd 2E	253 203 d 46
SRA A	CB 2F	203 47
SRA B	CB 28	203 40
SRA C	CB 29	203 41
SRA D	CB 2A	203 42
SRA E	CB 2B	203 43
SRA H	CB 2C	203 44
SRA L	CB 2D	203 45
SRL (HL)	CB 3E	203 62
SRL (IX+d)	DD CB dd 3E	221 203 d 62
SRL (IY+d)	FD CB dd 3E	253 203 d 62
SRL A	CB 3F	203 63
SRL B	CB 38	203 56
SRL C	CB 39	203 57
SRL D	CB 3A	203 58
SRL E	CB 3B	203 59
SRL H	CB 3C	203 60
SRL L	CB 3D	203 61
SUB (HL)	96	150
SUB (IX+d)	DD 96 dd	221 150 d
SUB (IY+d)	FD 96 dd	253 150 d
SUB A	97	151
SUB B	90	144
SUB C	91	145
SUB D	92	146
SUB E	93	147
SUB H	94	148
SUB L	95	149

Mnémonique	Code hexadécimal	Code décimal
SUB n	D6 nn	214 n
XOR (HL)	AE	174
XOR (IX+d)	DD AE dd	221 174 d
XOR (IY+d)	FD AE dd	253 174 d
XOR A	AF	175
XOR B	A8	168
XOR C	A9	169
XOR D	AA	170
XOR E	AB	171
XOR H	AC	172
XOR L	AD	173
XOR n	EE nn	238 n

Tableau de désassemblage

n = nombre d'un octet.
nn = nombre de deux octets (adresse 16 bits) sous format OMS, OPS.
d = index de déplacement (codé dd sur deux octets).
e = index de déplacement pour les instructions de branchements relatifs (codé **ee** sur deux octets).

Code hexadécimal	Code décimal	Mnémonique
00	0	NOP
01 nn nn	1 n n	LD BC,nn
02	2	LD (BC),A
03	3	INC BC
04	4	INC B
05	5	DEC B
06 nn	6 n	LD B,n
07	7	RLCA
08	8	EX AF,AF'
09	9	ADD HL,BC
0A	10	LD A,(BC)
0B	11	DEC BC
0C	12	INC C
0D	13	DEC C
0E nn	14 n	LD C,n
0F	15	RRCA
10 ee	16 e	DJNZ e
11 nn nn	17 n n	LD DE,nn
12	18	LD (DE),A
13	19	INC DE
14	20	INC D
15	21	DEC D
16 nn	22	LD D,n
17	23	RLA
18 ee	24 e	JR e
19	25	ADD HL,DE
1A	26	LD A,(DE)
1B	27	DEC DE
1C	28	INC E
1D	29	DEC E
1E nn	30 n n	LD E,n
1F	31	RRA
20 ee	32 e	JR NZ,e
21 nn nn	33 n n	LD HL,nn
22 nn nn	34 n n	LD (nn),HL
23	35	INC HL
24	36	INC H
25	37	DEC H

Code hexadécimal	Code décimal	Mnémonique
26 nn	38 n	LD H,n
27	39	DAA
28 ee	40 e	JR Z,e
29	41	ADD HL,HL
2A nn nn	42 n n	LD HL,(nn)
2B	43	DEC HL
2C	44	INC L
2D	45	DEC L
2E nn	46 n	LD L,n
2F	47	CPL
30 ee	48 e	JR NC,e
31 nn nn	49 n n	LD SP,nn
32 nn nn	50 n n	LD (nn),A
33	51	INC SP
34	52	INC (HL)
35	53	DEC (HL)
36 nn	54 n	LD (HL),n
37	55	SCF
38 ee	56 e	JR C,e
39	57	ADD HL,SP
3A nn nn	58 n n	LD A,(nn)
3B	59	DEC SP
3C	60	INC A
3D	61	DEC A
3E nn	62 n	LD A,n
3F	63	CCF
40	64	LD B,B
41	65	LD B,C
42	66	LD B,D
43	67	LD B,E
44	68	LD B,H
45	69	LD B,L
46	70	LD B,(HL)
47	71	LD B,A
48	72	LD C,B
49	73	LD C,C
4A	74	LD C,D
4B	75	LD C,E
4C	76	LD C,H
4D	77	LD C,L
4E	78	LD C,(HL)
4F	79	LD C,A
50	80	LD D,B
51	81	LD D,C
52	82	LD D,D
53	83	LD D,E

Code hexadécimal	Code décimal	Mnémonique
54	84	LD D,H
55	85	LD D,L
56	86	LD D,(HL)
57	87	LD D,A
58	88	LD E,B
59	89	LD E,C
5A	90	LD E,D
5B	91	LD E,E
5C	92	LD E,H
5D	93	LD E,L
5E	94	LD E,(HL)
5F	95	LD E,A
60	96	LD H,B
61	97	LD H,C
62	98	LD H,D
63	99	LD H,E
64	100	LD H,H
65	101	LD H,L
66	102	LD H,(HL)
67	103	LD H,A
68	104	LD L,B
69	105	LD L,C
6A	106	LD L,D
6B	107	LD L,E
6C	108	LD L,H
6D	109	LD L,L
6E	110	LD L,(HL)
6F	111	LD L,A
70	112	LD (HL),B
71	113	LD (HL),C
72	114	LD (HL),D
73	115	LD (HL),E
74	116	LD (HL),H
75	117	LD (HL),L
76	118	HALT
77	119	LD (HL),A
78	120	LD A,B
79	121	LD A,C
7A	122	LD A,D
7B	123	LD A,E
7C	124	LD A,H
7D	125	LD A,L
7E	126	LD A,(HL)
7F	127	LD A,A
80	128	ADD A,B

Code hexadécimal	Code décimal	Mnémonique
81	129	ADD A,C
82	130	ADD A,D
83	131	ADD A,E
84	132	ADD A,H
85	133	ADD A,L
86	134	ADD A,(HL)
87	135	ADD A,A
88	136	ADC A,B
89	137	ADC A,C
8A	138	ADC A,D
8B	139	ADC A,E
8C	140	ADC A,H
8D	141	ADC A,L
8E	142	ADC A,(HL)
8F	143	ADC A,A
90	144	SUB B
91	145	SUB C
92	146	SUB D
93	147	SUB E
94	148	SUB H
95	149	SUB L
96	150	SUB (HL)
97	151	SUB A
98	152	SBC A,B
99	153	SBC A,C
9A	154	SBC A,D
9B	155	SBC A,E
9C	156	SBC A,H
9D	157	SBC A,L
9E	158	SBC A,(HL)
9F	159	SBC A,A
A0	160	AND B
A1	161	AND C
A2	162	AND D
A3	163	AND E
A4	164	AND H
A5	165	AND L
A6	166	AND (HL)
A7	167	AND A
A8	168	XOR B
A9	169	XOR C
AA	170	XOR D
AB	171	XOR E
AC	172	XOR H
AD	173	XOR L
AE	174	XOR (HL)

Code hexadécimal	Code décimal	Mnémonique
AF	175	XOR A
B0	176	OR B
B1	177	OR C
B2	178	OR D
B3	179	OR E
B4	180	OR H
B5	181	OR L
B6	182	OR (HL)
B7	183	OR A
B8	184	CP B
B9	185	CP C
BA	186	CP D
BB	187	CP E
BC	188	CP H
BD	189	CP L
BE	190	CP (HL)
BF	191	CP A
C0	192	RET NZ
C1	193	POP BC
C2 nn nn	194 n n	JP NZ,nn
C3 nn nn	195 n n	JP nn
C4 nn nn	196 n n	CALL NZ,nn
C5	197	PUSH BC
C6 nn	198 n	ADD A,n
C7	199	RST 00H
C8	200	RET Z
C9	201	RET
CA nn nn	202 n n	JP Z,nn
CB 00	203 0	RLC B
CB 01	203 1	RLC C
CB 02	203 2	RLC D
CB 03	203 3	RLC E
CB 04	203 4	RLC H
CB 05	203 5	RLC L
CB 06	203 6	RLC (HL)
CB 07	203 7	RLC A
CB 08	203 8	RRC B
CB 09	203 9	RRC C
CB 0A	203 10	RRC D
CB 0B	203 11	RRC E
CB 0C	203 12	RRC H
CB 0D	203 13	RRC L
CB 0E	203 14	RRC (HL)
CB 0F	203 15	RRC A
CB 10	203 16	RL B
CB 11	203 17	RL C

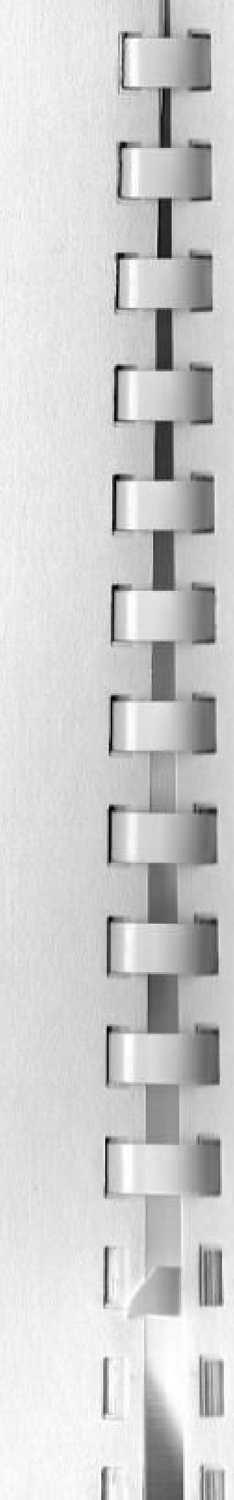
Code hexadécimal	Code décimal	Mnémonique
CB 12	203 18	RL D
CB 13	203 19	RL E
CB 15	203 20	RL L
CB 16	203 22	RL (HL)
CB 17	203 23	RL A
CB 18	203 24	RR B
CB 19	203 25	RR C
CB 1A	203 26	RR D
CB 1B	203 27	RR E
CB 1C	203 28	RR H
CB 1D	203 29	RR L
CB 1E	203 30	RR (HL)
CB 1F	203 31	RR A
CB 20	203 32	SLA B
CB 21	203 33	SLA C
CB 22	203 34	SLA D
CB 23	203 35	SLA E
CB 24	203 36	SLA H
CB 25	203 37	SLA L
CB 26	203 38	SLA (HL)
CB 27	203 39	SLA A
CB 28	203 40	SRA B
CB 29	203 41	SRA C
CB 2A	203 42	SRA D
CB 2B	203 43	SRA E
CB 2C	203 44	SRA H
CB 2D	203 45	SRA L
CB 2E	203 46	SRA (HL)
CB 2F	203 47	SRA A
CB 38	203 56	SRL B
CB 39	203 57	SRL C
CB 3A	203 58	SRL D
CB 3B	203 59	SRL E
CB 3C	203 60	SRL H
CB 3D	203 61	SRL L
CB 3E	203 62	SRL (HL)
CB 3F	203 63	SRL A
CB 40	203 64	BIT 0,B
CB 41	203 65	BIT 0,C
CB 42	203 66	BIT 0,D
CB 43	203 67	BIT 0,E
CB 44	203 68	BIT 0,H
CB 45	203 69	BIT 0,L
CB 46	203 70	BIT 0,(HL)
CB 47	203 71	BIT 0,A
CB 48	203 72	BIT 1,B

Code hexadécimal	Code décimal	Mnémonique
CB 49	203 73	BIT 1,C
CB 4A	203 74	BIT 1,D
CB 4B	203 75	BIT 1,E
CB 4C	203 76	BIT 1,H
CB 4D	203 77	BIT 1,L
CB 4E	203 78	BIT 1,(HL)
CB 4F	203 79	BIT 1,A
CB 50	203 80	BIT 2,B
CB 51	203 81	BIT 2,C
CB 52	203 82	BIT 2,D
CB 53	203 83	BIT 2,E
CB 54	203 84	BIT 2,H
CB 55	203 85	BIT 2,L
CB 56	203 86	BIT 2,(HL)
CB 57	203 87	BIT 2,A
CB 58	203 88	BIT 3,B
CB 59	203 89	BIT 3,C
CB 5A	203 90	BIT 3,D
CB 5B	203 91	BIT 3,E
CB 5C	203 92	BIT 3,H
CB 5D	203 93	BIT 3,L
CB 5E	203 94	BIT 3,(HL)
CB 5F	203 95	BIT 3,A
CB 60	203 96	BIT 4,B
CB 61	203 97	BIT 4,C
CB 62	203 98	BIT 4,D
CB 63	203 99	BIT 4,E
CB 64	203 100	BIT 4,H
CB 65	203 101	BIT 4,L
CB 66	203 102	BIT 4,(HL)
CB 67	203 103	BIT 4,A
CB 68	203 104	BIT 5,B
CB 69	203 105	BIT 5,C
CB 6A	203 106	BIT 5,D
CB 6B	203 107	BIT 5,E
CB 6C	203 108	BIT 5,H
CB 6D	203 109	BIT 5,L
CB 6E	203 110	BIT 5,(HL)
CB 6F	203 111	BIT 5,A
CB 70	203 112	BIT 6,B
CB 71	203 113	BIT 6,C
CB 72	203 114	BIT 6,D
CB 73	203 115	BIT 6,E
CB 74	203 116	BIT 6,H
CB 75	203 117	BIT 6,L
CB 76	203 118	BIT 6,(HL)

Code hexadécimal	Code décimal	Mnémonique
CB 77	203 119	BIT 6,A
CB 78	203 120	BIT 7,B
CB 79	203 121	BIT 7,C
CB 7A	203 122	BIT 7,D
CB 7B	203 123	BIT 7,E
CB 7C	203 124	BIT 7,H
CB 7D	203 125	BIT 7,L
CB 7E	203 126	BIT 7,(HL)
CB 7F	203 127	BIT 7,A
CB 80	203 128	RES 0,B
CB 81	203 129	RES 0,C
CB 82	203 130	RES 0,D
CB 83	203 131	RES 0,E
CB 84	203 132	RES 0,H
CB 85	203 133	RES 0,L
CB 86	203 134	RES 0,(HL)
CB 87	203 135	RES 0,A
CB 88	203 136	RES 1,B
CB 89	203 137	RES 1,C
CB 8A	203 138	RES 1,D
CB 8B	203 139	RES 1,E
CB 8C	203 140	RES 1,H
CB 8D	203 141	RES 1,L
CB 8E	203 142	RES 1,(HL)
CB 8F	203 143	RES 1,A
CB 90	203 144	RES 2,B
CB 91	203 145	RES 2,C
CB 92	203 146	RES 2,D
CB 93	203 147	RES 2,E
CB 94	203 148	RES 2,H
CB 95	203 149	RES 2,L
CB 96	203 150	RES 2,(HL)
CB 97	203 151	RES 2,A
CB 98	203 152	RES 3,B
CB 99	203 153	RES 3,C
CB 9A	203 154	RES 3,D
CB 9B	203 155	RES 3,E
CB 9C	203 156	RES 3,H
CB 9D	203 157	RES 3,L
CB 9E	203 158	RES 3,(HL)
CB 9F	203 159	RES 3,A
CB A0	203 160	RES 4,B
CB A1	203 161	RES 4,C
CB A2	203 162	RES 4,D
CB A3	203 163	RES 4,E
CB A4	203 164	RES 4,H

Code hexadécimal	Code décimal	Mnémonique
CB A5	203 165	RES 4,L
CB A6	203 166	RES 4,(HL)
CB A7	203 167	RES 4,A
CB A8	203 168	RES 5,B
CB AA	203 170	RES 5,D
CB AB	203 171	RES 5,E
CB AC	203 172	RES 5,H
CB AD	203 173	RES 5,L
CB AE	203 174	RES 5,(HL)
CB AF	203 175	RES 5,A
CB B0	203 176	RES 6,B
CB B1	203 177	RES 6,C
CB B2	203 178	RES 6,D
CB B3	203 179	RES 6,E
CB B4	203 180	RES 6,H
CB B5	203 181	RES 6,L
CB B6	203 182	RES 6,(HL)
CB B7	203 183	RES 6,A
CB B8	203 184	RES 7,B
CB B9	203 185	RES 7,C
CB BA	203 186	RES 7,D
CB BB	203 187	RES 7,E
CB BC	203 188	RES 7,H
CB BD	203 189	RES 7,L
CB BE	203 190	RES 7,(HL)
CB BF	203 191	RES 7,A
CB C0	203 192	SET 0,B
CB C1	203 193	SET 0,C
CB C2	203 194	SET 0,D
CB C3	203 195	SET 0,E
CB C4	203 196	SET 0,H
CB C5	203 197	SET 0,L
CB C6	203 198	SET 0,(HL)
CB C7	203 199	SET 0,A
CB C8	203 200	SET 1,B
CB C9	203 201	SET 1,C
CB CA	203 202	SET 1,D
CB CB	203 203	SET 1,E
CB CC	203 204	SET 1,H
CB CD	203 205	SET 1,L
CB CE	203 206	SET 1,(HL)
CB CF	203 207	SET 1,A
CB D0	203 208	SET 2,B
CB D1	203 209	SET 2,C
CB D2	203 210	SET 2,D
CB D3	203 211	SET 2,E

Code hexadécimal	Code décimal	Mnémonique
CB D4	203 212	SET 2,H
CB D5	203 213	SET 2,L
CB D6	203 214	SET 2,(HL)
CB D7	203 215	SET 2,A
CB D8	203 216	SET 3,B
CB D9	203 217	SET 3,C
CB DA	203 218	SET 3,D
CB DB	203 219	SET 3,E
CB DC	203 220	SET 3,H
CB DD	203 221	SET 3,L
CB DE	203 222	SET 3,(HL)
CB DF	203 223	SET 3,A
CB E0	203 224	SET 4,B
CB E1	203 225	SET 4,C
CB E2	203 226	SET 4,D
CB E3	203 227	SET 4,E
CB E4	203 228	SET 4,H
CB E5	203 229	SET 4,L
CB E6	203 230	SET 4,(HL)
CB E7	203 231	SET 4,A
CB E8	203 232	SET 5,B
CB E9	203 233	SET 5,C
CB EA	203 234	SET 5,D
CB EB	203 235	SET 5,E
CB EC	203 236	SET 5,H
CB ED	203 237	SET 5,L
CB EE	203 238	SET 5,(HL)
CB EF	203 239	SET 5,A
CB F0	203 240	SET 6,B
CB F1	203 241	SET 6,C
CB F2	203 242	SET 6,D
CB F3	203 243	SET 6,E
CB F4	203 244	SET 6,H
CB F5	203 245	SET 6,L
CB F6	203 246	SET 6,(HL)
CB F7	203 247	SET 6,A
CB F8	203 248	SET 7,B
CB F9	203 249	SET 7,C
CB FA	203 250	SET 7,D
CB FB	203 251	SET 7,E
CB FC	203 252	SET 7,H
CB FD	203 253	SET 7,L
CB FE	203 254	SET 7,(HL)
CB FF	203 255	SET 7,A
CC nn nn	204 n n	CALL Z,nn
CD nn nn	205 n n	CALL nn



Code hexadécimal	Code décimal	Mnémonique
CE nn	206 n	ADC A,n
CF	207	RST 08H
D0	208	RET NC
D1	209	POP DE
D2 nn nn	210 n n	JP NC,nn
D3 nn	211 n	OUT n,A
D4 nn nn	212 n n	CALL NC,nn
D5	213	PUSH DE
D6 nn	214 n	SUB n
D7	215	RST 10H
D8	216	RET C
D9	217	EXX
DA nn nn	218 n n	JP C,nn
DB nn	219 n	IN A,n
DC nn nn	220 n n	CALL C,nn
DD 09	221 9	ADD IX,BC
DD 19	221 25	ADD IX,DE
DD 21 nn nn	221 33 n n	LD IX,nn
DD 22 nn nn	221 34 n n	LD (nn),IX
DD 23	221 35	INC IX
DD 29	221 41	ADD IX,IX
DD 2A nn nn	221 42 n n	LD IX,(nn)
DD 2B	221 43	DEC IX
DD 34 dd	221 52 d	INC (IX+d)
DD 35 dd	221 53 d	DEC (IX+d)
DD 36 dd nn	221 54 d n	LD (IX+d),n
DD 39	221 57	ADD IX,SP
DD 46 dd	221 70 d	LD B,(IX+d)
DD 4E dd	221 78 d	LD C,(IX+d)
DD 56 dd	221 86 d	LD D,(IX+d)
DD 5E dd	221 94 d	LD E,(IX+d)
DD 66 dd	221 102 d	LD H,(IX+d)
DD 6E dd	221 110 d	LD L,(IX+d)
DD 70 dd	221 112 d	LD (IX+d),B
DD 71 dd	221 113 d	LD (IX+d),C
DD 72 dd	221 114 d	LD (IX+d),D
DD 73 dd	221 115 d	LD (IX+d),E
DD 74 dd	221 116 d	LD (IX+d),H
DD 75 dd	221 117 d	LD (IX+d),L
DD 77 dd	221 119 d	LD (IX+d),A
DD 7E dd	221 126 d	LD A,(IX+d)
DD 86 dd	221 134 d	ADD A,(IX+d)
DD 8E dd	221 142 d	ADC A,(IX+d)
DD 96 dd	221 150 d	SUB (IX+d)
DD 9E dd	221 158 d	SBC A,(IX+d)
DD A6 dd	221 166 d	AND (IX+d)

MICROPROCESSEUR Z80

Code hexadécimal	Code décimal	Mnémonique
DD AE dd	221 174 d	XOR (IX+d)
DD B6 dd	221 182 d	OR (IX+d)
DD BE dd	221 190 d	CP (IX+d)
DD CB dd 06	221 203 d 6	RLC (IX+d)
DD CB dd 0E	221 203 d 14	RRC (IX+d)
DD CB dd 16	221 203 d 22	RL (IX+d)
DD CB dd 1E	221 203 d 30	RR (IX+d)
DD CB dd 26	221 203 d 38	SLA (IX+d)
DD CB dd 2E	221 203 d 46	SRA (IX+d)
DD CB dd 3E	221 203 d 62	SRL (IX+d)
DD CB dd 46	221 203 d 70	BIT 0,(IX+d)
DD CB dd 4E	221 203 d 78	BIT 1,(IX+d)
DD CB dd 56	221 203 d 86	BIT 2,(IX+d)
DD CB dd 5E	221 203 d 94	BIT 3,(IX+d)
DD CB dd 66	221 203 d 102	BIT 4,(IX+d)
DD CB dd 6E	221 203 d 110	BIT 5,(IX+d)
DD CB dd 76	221 203 d 118	BIT 6,(IX+d)
DD CB dd 7E	221 203 d 126	BIT 7,(IX+d)
DD CB dd 86	221 203 d 134	RES 0,(IX+d)
DD CB dd 8E	221 203 d 142	RES 1,(IX+d)
DD CB dd 96	221 203 d 150	RES 2,(IX+d)
DD CB dd 9E	221 203 d 158	RES 3,(IX+d)
DD CB dd A6	221 203 d 166	RES 4,(IX+d)
DD CB dd AE	221 203 d 174	RES 5,(IX+d)
DD CB dd B6	221 203 d 182	RES 6,(IX+d)
DD CB dd BE	221 203 d 190	RES 7,(IX+d)
DD CB dd C6	221 203 d 198	SET 0,(IX+d)
DD CB dd CE	221 203 d 206	SET 1,(IX+d)
DD CB dd D6	221 203 d 214	SET 2,(IX+d)
DD CB dd DE	221 203 d 222	SET 3,(IX+d)
DD CB dd E6	221 203 d 230	SET 4,(IX+d)
DD CB dd EE	221 203 d 238	SET 5,(IX+d)
DD CB dd F6	221 203 d 246	SET 6,(IX+d)
DD CB dd FE	221 203 d 254	SET 7,(IX+d)
DD E1	221 225	POP IX
DD E3	221 227	EX (SP),IX
DD E5	221 229	PUSH IX
DD E9	221 233	JP (IX)
DD F9	221 249	LD SP,IX
DE nn	222 n	SBC A,n
DF	223	RST 18H
E0	224	RET PO
E1	225	POP HL
E2 nn nn	226 n n	JP PO,nn
E3	227	EX (SP),HL
E4 nn nn	228 n n	CALL PO,nn

Code hexadécimal	Code décimal	Mnémonique
E5	229	PUSH HL
E6 nn	230 n	AND n
E7	231	RST 20H
E8	232	RET PE
E9	233	JP (HL)
EA nn nn	234 n n	JP PE,nn
EB	235	EX DE,HL
EC nn nn	236 n n	CALL PE,nn
ED 40	237 64	IN B,(C)
ED 41	237 65	OUT (C),B
ED 42	237 66	SBC HL,BC
ED 43 nn nn	237 67 n n	LD (nn),BC
ED 44	237 68	NEG
ED 45	237 69	RETN
ED 46	237 70	IM 0
ED 47	237 71	LD I,A
ED 48	237 72	IN C,(C)
ED 49	237 73	OUT (C),C
ED 4A	237 74	ADC HL,BC
ED 4B nn nn	237 75 n n	LD BC,(nn)
ED 4D	237 77	RETI
ED 4F	237 79	LD R,A
ED 50	237 80	IN D,(C)
ED 51	237 81	OUT (C),D
ED 52	237 82	SBC HL,DE
ED 53 nn nn	237 83 n n	LD (nn),DE
ED 56	237 86	IM 1
ED 57	237 87	LD A,I
ED 58	237 88	IN E,(C)
ED 59	237 89	OUT (C),E
ED 5A	237 90	ADC HL,DE
ED 5B nn nn	237 91 n n	LD DE,(nn)
ED 5E	237 94	IM 2
ED 5F	237 95	LD A,R
ED 60	237 96	IN H,(C)
ED 61	237 97	OUT (C),H
ED 62	237 98	SBC HL,HL
ED 67	237 103	RRD
ED 68	237 104	IN L,(C)
ED 69	237 105	OUT (C),L
ED 6A	237 106	ADC HL,HL
ED 6F	237 111	RLD
ED 72	237 114	SBC HL,SP
ED 73 nn nn	237 115 n n	LD (nn),SP
ED 78	237 120	IN A,(C)
ED 79	237 121	OUT (C),A

Code hexadécimal	Code décimal	Mnémonique
ED 7A	237 122	ADC HL,SP
ED 7B nn nn	237 123 n n	LD SP,(nn)
ED A0	237 160	LDI
ED A1	237 161	CPI
ED A2	237 162	INI
ED A3	237 163	OUTI
ED A8	237 168	LDD
ED A9	237 169	CPD
ED AA	237 170	IND
ED AB	237 171	OUTD
ED B0	237 176	LDIR
ED B1	237 177	CPIR
ED B2	237 178	INIR
ED B3	237 179	OTIR
ED B8	237 184	LDDR
ED B9	237 185	CPDR
ED BA	237 186	INDR
ED BB	237 187	OTDR
EE nn	238 n	XOR n
EF	239	RST 28H
F0	240	RET P
F1	241	POP AF
F2 nn nn	242 n n	JP P,nn
F3	243	DI
F4 nn nn	244 n n	CALL P,nn
F5	245	PUSH AF
F6 nn	246 n	OR n
F7	247	RST 30H
F8	248	RET M
F9	249	LD SP,HL
FA nn nn	250 n n	JP M,nn
FB	251	EI
FC nn nn	252 n n	CALL M,nn
FD 09	253 9	ADD IY,BC
FD 19	253 25	ADD IY,DE
FD 21 nn nn	253 33 n n	LD IY,nn
FD 22 nn nn	253 34 n n	LD (nn),IY
FD 23	253 35	INC IY
FD 29	253 41	ADD IY,IY
FD 2A nn nn	253 42 n n	LD IY,(nn)
FD 2B	253 43	DEC IY
FD 34 dd	253 52 d	INC (IY+d)
FD 35 dd	253 53 d	DEC (IY+d)
FD 36 dd nn	253 54 d n	LD (IY+d),n
FD 39	253 57	ADD IY,SP
FD 46 dd	253 70 d	LD B,(IY+d)

Code hexadécimal	Code décimal	Mnémonique
FD 4E dd	253 78 d	LD C,(IY+d)
FD 56 dd	253 86 d	LD D,(IY+d)
FD 5E dd	253 94 d	LD E,(IY+d)
FD 66 dd	253 102 d	LD H,(IY+d)
FD 6E dd	253 110 d	LD L,(IY+d)
FD 70 dd	253 112 d	LD (IY+d),B
FD 71 dd	253 113 d	LD (IY+d),C
FD 72 dd	253 114 d	LD (IY+d),D
FD 73 dd	253 115 d	LD (IY+d),E
FD 74 dd	253 116 d	LD (IY+d),H
FD 75 dd	253 117 d	LD (IY+d),L
FD 77 dd	253 119 d	LD (IY+d),A
FD 7E dd	253 126 d	LD A,(IY+d)
FD 86 dd	253 134 d	ADD A,(IY+d)
FD 8E dd	253 142 d	ADC A,(IY+d)
FD 96 dd	253 150 d	SUB (IY+d)
FD 9E dd	253 158 d	SBC A,(IY+d)
FD A6 dd	253 166 d	AND (IY+d)
FD AE dd	253 174 d	XOR (IY+d)
FD B6 dd	253 182 d	OR (IY+d)
FD BE dd	253 190 d	CP (IY+d)
FD CB dd 06	253 203 d 6	RLC (IY+d)
FD CB dd 0E	253 203 d 14	RRC (IY+d)
FD CB dd 16	253 203 d 22	RL (IY+d)
FD CB dd 1E	253 203 d 30	RR (IY+d)
FD CB dd 26	253 203 d 38	SLA (IY+d)
FD CB dd 2E	253 203 d 46	SRA (IY+d)
FD CB dd 3E	253 203 d 62	SRL (IY+d)
FD CB dd 46	253 203 d 70	BIT 0,(IY+d)
FD CB dd 4E	253 203 d 78	BIT 1,(IY+d)
FD CB dd 56	253 203 d 86	BIT 2,(IY+d)
FD CB dd 5E	253 203 d 94	BIT 3,(IY+d)
FD CB dd 66	253 203 d 102	BIT 4,(IY+d)
FD CB dd 6E	253 203 d 110	BIT 5,(IY+d)
FD CB dd 76	253 203 d 118	BIT 6,(IY+d)
FD CB dd 7E	253 203 d 126	BIT 7,(IY+d)
FD CB dd 86	253 203 d 134	RES 0,(IY+d)
FD CB dd 8E	253 203 d 142	RES 1,(IY+d)
FD CB dd 96	253 203 d 150	RES 2,(IY+d)
FD CB dd 9E	253 203 d 158	RES 3,(IY+d)
FD CB dd A6	253 203 d 166	RES 4,(IY+d)
FD CB dd AE	253 203 d 174	RES 5,(IY+d)
FD CB dd B6	253 203 d 182	RES 6,(IY+d)
FD CB dd BE	253 203 d 190	RES 7,(IY+d)
FD CB dd C6	253 203 d 198	SET 0,(IY+d)
FD CB dd CE	253 203 d 206	SET 1,(IY+d)

Code hexadécimal	*Code décimal*	*Mnémonique*
FD CB dd D6	253 203 d 214	SET 2,(IY+d)
FD CB dd DE	253 203 d 222	SET 3,(IY+d)
FD CB dd E6	253 203 d 230	SET 4,(IY+d)
FD CB dd EE	253 203 d 238	SET 5,(IY+d)
FD CB dd F6	253 203 d 246	SET 6,(IY+d)
FD CB dd FE	253 203 d 254	SET 7,(IY+d)
FD E1	253 225	POP IY
FD E3	253 227	EX (SP),IY
FD E5	253 229	PUSH IY
FD E9	253 233	JP (IY)
FD F9	253 249	LD SP,IY
FE nn	254 n	CP n
FF	255	RST 38H

ADRESSES ROM

ROM BASIC

0000H-7FFFH
0D-32767D
Longueur = 32K octets

Structure

Bas de la mémoire → 0000H — 0D

0000H	SYSTEME D'EXPLOITATION (Moniteur Système) BIOS + DRIVERS DE PERIPHERIQUE + TABLES INTERNES	0D	16K octets
4000H	INTERPRETEUR BASIC (routines d'exécution des mots-clés)	16384D	16K octets
8000H	DEBUT ZONE RAM BASIC (TIP)	32768D	

Vers haut de la mémoire ↓

Rôle

La mémoire morte (ROM) de 32 Ko du MSX comporte deux parties égales de 16 Ko.

ZONE 1 (ROM1) - (0000H-3FFFH) : contient le **système d'exploitation.**

Elle est essentiellement constituée de :

- vecteurs RST + vecteurs BIOS (0000H-015CH) ;
- drivers de périphériques + processeurs spécialisés (cassette écran imprimante VDP, PSG, PPI) ;
- éditeur d'écran Basic ;
- routines arithmétiques et mathématiques (268CH-392DH) ;
- tables internes du système (392EH-3FD1H) ;
- messages affichés à l'écran (3FD2H-3FE1H).

ZONE 2 (ROM2) - (4000H-7FFFH) : contient l'**interpréteur Basic.**

C'est-à-dire l'ensemble des routines d'analyse et d'exécution des mots-clés (à part opérateur et fonction mathématiques en ROM)

0000H-3FFFH
0D-16383D
Longueur = 16K octets

Adresse Accès DEC	*Adresse Accès HEX*	*Rôle*
0	0000 (1)-(2)	Point d'entrée de l'initialisation (CHKRAM).
8	0008 (2)	RST 8 (SYNCHK) : cette routine regarde le caractère courant pointé par HL et le compare au caractère qui suit le RST, si le caractère n'est pas celui qui est attendu, la routine d'impression de "SYNTAX ERROR" est appelée, sinon elle saute le caractère et revient. Tous les registres sont préservés, exceptés AF et HL qui sont modifiés.
16	0010 (2)	RST 10 (CHRGTR) : cette routine utilise HL comme pointeur et charge dans A le caractère pointé par HL, elle positionne les drapeaux du registre F en fonction du type du caractère dans A : si le caractère est numérique, C (carry) est positionné ; si le caractère est : (multi-instruction) ou fin de ligne, il positionne l'indicateur de zéro (Z). Tous les registres sont préservés, excepté AF et HL qui sont modifiés.
24	0018 (2)	RST 18 (OUTDO) : cette routine sort le caractère contenu dans A sur le périphérique (CRT ou LPT) déterminé par PRTFLG. Les registres ne sont pas modifiés.
32	0020 (2)	RST 20 (DCOMPR) : cette routine compare le contenu de HL avec le contenu de DE ; si HL = DE, l'indicateur de zéro (Z) est positionné, si HL< DE, l'indicateur carry (C) est positionné.

Voir table des vecteurs du BIOS

(1) Contient, comme premier octet, ***F3****, code machine pour* ***DI*** *(Disable Interrupt) → interdiction des interruptions.*
(2) RST n = appel d'une routine à l'adresse n en mémoire basse (page 0) en n'utilisant qu'un seul octet en code machine. (RST 10H:D7, RST 18H:DF, RST 20H:E7, RST 28H:EF, RST 30H:F7, RST 38H:FF, RST 08H:CF, RST 00H:C7).

Adresse Accès DEC	Adresse Accès HEX	Rôle
40	0028 (2)	RST 28 : test de l'Indicateur de Type de donné (3) (F663H) et positionne les indicateurs du registre F. SIGNE=ENTIER, ZERO=CHAINE, PARITE=SIMPLE PRECISION, NO CARRY=DOUBLE PRECISION (GETYPR).
48	0030 (2)	RST 30 : gestion des SLOTS (CALLF).
56	0038 (2)	RST 38 : gestion des interruptions (KEYINT).
59	003B	Début de la table des JUMPS BIOS. Cette table se termine à l'adresse 015CH.
438	01B6	Routines de gestion des SLOTS mémoires. Ces routines se terminent en 02D6H.
727	02D7	Suite de l'initialisation (adresse 0).
1019	03FB	Processus de traitement du BREAK (CTRL-C).
1181	049D	Initialisation du PSG.
1294	050E	Initialisation mode SCREEN 0 (VDP : TEXTE).
1339	053B	Initialisation mode SCREEN 1 (VDP : GRAPHIQUE I).
1490	05D2	Initialisation mode SCREEN 2 (VDP : GRAPHIQUE II).
1567	061F	Initialisation mode SCREEN 3 (VDP : MULTICOLORE).
1764	06E4	Place dans HL l'adresse d'un SPRITE dans la TGS si A contient son numéro.
1785	06F9	Place dans HL l'adresse d'un SPRITE dans la TAS si A contient son numéro.
1786	0704	Place dans A le nombre 8 si les SPRITES sont au format 8 x 8 et 32 si les SPRITES sont au format 16 x 16.
1860	0744	Ecriture dans le VDP, DE pointe vers le texte à écrire, BC contient la longueur du texte et HL pointe vers l'adresse de la VIDEORAM où le texte doit être écrit.

(Lignes 40 à 59 : Voir table des vecteurs du BIOS)

(2) RSTU = appel d'une routine à l'adresse n en mémoire basse (page 0) en n'utilisant qu'un seul octet en code machine. (RST 10H:D7, RST 18H:DF, RST 20H:E7, RST 28H:EF, RST 30H:F7, RST 38H:FF, RST 08H:CF, RST 00H:C7).
(3) ITD en abrégé.

Adresse Accès DEC	Adresse Accès HEX	Rôle
1997	07CD	Ecriture dans la VIDEORAM, A contient la valeur à écrire et HL pointe vers l'adresse de la VIDEORAM.
2007	07D7	Lecture de la VIDEORAM, HL contient l'adresse à lire, au retour A contient la valeur lue.
2015	07DF	Positionne la VIDEORAM à l'adresse contenue dans HL en vue d'une écriture.
2028	07EC	Comme ci-dessus, mais en vue d'une lecture.
2069	0815	Ecriture d'un même caractère plusieurs fois dans la VIDEORAM, HL contient l'adresse de la VIDEORAM, A contient le caractère et BC contient le nombre de fois qu'il faut écrire le caractère.
2107	083B	Retour à l'ancien mode TEXTE (40 ou 32) à la fin d'un programme ou à la suite d'une erreur.
2120	0848	CLS.
2127	084F	Initialise dans le mode SCREEN 0, 1, 2 ou 3 suivant la valeur de A.
2141	085D	Routine d'impression du contenu de A sur l'imprimante.
2180	0884	Routine de test de l'état de l'imprimante, si l'imprimante est BUSY, l'indicateur Z est positionné.
2190	088E	Positionnement du curseur en absolu suivant la séquence ESC Y colonne ligne, L contient la colonne et H la ligne.
2205	089D	Traitement du caractère à imprimer pour le rendre compatible avec les imprimantes non MSX.
2236	08BC	Affichage d'un caractère à l'écran.
2351	092F (1)	Table des valeurs des caractères spéciaux avec leurs adresses de traitement. Cette table se termine en 097FH.
2541	09ED	Affichage du curseur.
2599	0A27	Effacement du curseur.

(1) Voir pour plus de détail "Table des valeurs et des pointeurs d'adresse de traitement des codes de contrôle".

Adresse Accès DEC	Adresse Accès HEX	Rôle
2628	0A44	Déplacement du curseur à droite.
2636	0A44	BACKSPACE.
2647	0A57	Déplacement du curseur vers le haut.
2651	0A5B	Avance du curseur.
2657	0A61	Déplacement du curseur vers le bas.
2673	0A71	TABULATION.
2687	0A7F	HOME.
2693	0A85	Effacement de ligne.
2740	0AB4	Insertion de ligne.
2787	0AE3	Effacement du caractère précédent.
2812	0AFC	Effacement total de la ligne.
2814	0AFE	Effacement depuis la position du curseur jusqu'à la fin de la ligne.
2821	0B05	Effacement depuis la position du curseur jusqu'à la fin de la page.
2837	0B15	Effacement des touches de fonction.
2859	0B2B	Affichage des touches de fonction.
3132	0C3C	Routine de traitement de l'interruption hardware générée par le VDP.
3346	0D12	Test de touche clavier enfoncé.
3434	0D6A	Lecture d'un caractère en provenance du clavier. Cette routine réalise une seule scrutation sans attente et sans bouclage.
3493	0DA5	Table de transcodage du clavier. Cette table se termine en 0EC4H.
3846	0F06	Traitement de la touche HOME-CLS.
3901	0F3D	Eteint ou allume le témoin CAPS en fonction du contenu de A.
3910	0F46	Traitement de la touche STOP.
3962	0F7A	Positionne le bit 7 du PORT C du PPI en fonction de la valeur de A, ce bit permet des effets sonores.
4129	1021	Routine de codage de la touche enfoncée.

Adresse Accès DEC	Adresse Accès HEX	Rôle
4147	1033	Table de codage clavier. Cette table se termine en 10C1H.
4290	10C2	Mise à jour du pointeur dans le buffer du clavier.
4299	10CB	Routine de saisie d'un caractère.
4354	1102	Routine d'écriture dans le PSG : A contient le numéro du registre et E contient la valeur à écrire.
4364	110C	Routine de lecture du registre 14 : PORT A du PSG. Au retour, A contient la valeur lue.
4366	110E	Routine de lecture d'un registre du PSG. A l'appel, A contient le numéro du registre, au retour, A contient la valeur lue.
4371	1113	BEEP : émission du BEEP. Cette routine modifie tous les registres.
4411	113B	Cette routine lit les informations sur l'action en cours dans la file musicale (MUSIC QUEUE). En entrée, A contient le numéro du canal (0 à 2). Cette routine est utilisée essentiellement par l'instruction PLAY.
4464	1170	Routine de positionnement de la fréquence pour l'instruction PLAY.
4481	1181	Routine de positionnement du volume pour l'instruction PLAY.
4501	1195	Routine de positionnement de la période d'enveloppe pour l'instruction PLAY.
4590	11EE	Routine de lecture des manettes de jeux ou des flèches du clavier. En entrée, A contient 0 pour le clavier, 1 ou 2 pour la manette correspondante.
4691	1253	Routine de lecture des boutons de tir ou de la barre d'espace en fonction de la valeur de A.
4723	1273	Routine de lecture de manette analogique (PDL).
4780	12AC	Routine de lecture de la tablette graphique (PAD).

Adresse Accès DEC	Adresse Accès HEX	Rôle
4996	1384	Positionnement du relais de la cassette (MOTOR).
5039	13A9 (1)	Table des valeurs par défaut pour les touches de fonction (F1-F10). Cette table se termine en 1448H.
5193	1449	Lecture du registre d'état du VDP.
5196	144C	Lecture du port A du PPI.
5202	1452	Lecture d'une ligne clavier. En entrée, A contient le numéro de la ligne à lire, en sortie, A contient la valeur lue.
5215	145F	Routine de test de présence de fichier.
5226	146A	Comparaison de DE avec HL (RST 20).
5232	1470	Routine de positionnement dans la file musicale. A contient le numéro de la voix.
5236	1474	Idem 1470, mais le numéro de voix se trouve en FB38H et L doit être positionné en entrée sur la valeur du déplacement.
5239	1477	Idem 1474, mais A doit contenir le numéro de voix.
5266	1492	A cette adresse, commence la première routine de manipulation des FILES (QUEUES). Les FILES peuvent être de longueur : 2^1-1 jusqu'à 2^8-1. Une FILE peut être initialisée à n'importe quel moment et n'importe où. Un pointeur fournit l'adresse de la table des FILES. La table des FILES contient toutes les informations sur chaque FILE. Ces informations sont représentées par six octets. Le premier donne l'OFFSET pour une mise en FILE, le second donne l'OFFSET pour une prise en FILE, le troisième contient le premier caractère de la FILE, le quatrième la longueur de la FILE et le couple cinquième-sixième, l'adresse de la FILE. Toutes les routines supposent que A contient le numéro de la FILE et que F3F3H contient l'adresse de la table des FILES.

(1) Voir pour plus de détail : "Table des valeurs par défaut des touches de fonction".

Adresse Accès DEC	Adresse Accès HEX	Rôle
5281	14A1	Routine de mise en fin de FILE. Le caractère contenu dans E est mis en FILE, si la FILE est pleine, l'indicateur Z est positionné.
5293	14AD	Routine de prise en début de FILE. Le caractère est mis dans A, l'indicateur Z est positionné si la file est vide.
5329	14D1	Routine d'écriture du caractère contenu dans E en début de FILE.
5338	14DA	Initialise une FILE à vide. B=longueur de la FILE, (DE)=adresse.
5355	14EB	Routine qui retourne dans A le nombre d'octets libres dans la FILE.
5391	150F	Fin des routines de FILES.
5392	1510	Ecriture d'un caractère sur l'écran en mode graphique.
5529	1599	Routine d'ajustage des valeurs de X et Y. En entrée, BC contient X et DE contient Y. En sortie, ces registres contiennent les mêmes valeurs mais ajustées (MODULO).
5593	15D9	Test de la valeur courante de SCREEN.
5599	15DF	Routine de détermination de l'adresse de la VIDEORAM en fonction de la valeur de X et de Y. En entrée, BC contient X et DE contient Y. En sortie, HL contient l'adresse de la VIDEORAM et A contient le masque à appliquer.
5643	160B (1)	Table des puissances de 2. Fin en 1612H.
5689	1639	Lecture de l'accumulateur graphique : F92AH contient la localisation dans la VIDEORAM qui est transférée dans HL et F92CH contient le masque qui est transféré dans A.
5696	1640	Ecriture de l'accumulateur graphique (voir ci-dessus).
5703	1647	Lit les attributs de l'accumulateur graphique courant.

(1) contient : 80H (160BH) - 40H - 20H - 10H - 08H - 04H - 02H - 01H (1612H)
128D - 64D - 32D - 16D - 8D - 4D - 2D - 1D
(2^7 - 2^6 - 2^5 - 2^4 - 2^3 - 2^2 - 2^1 - 2^0)

ADRESSES ROM

Adresse Accès DEC	Adresse Accès HEX	Rôle
5750	1676	Positionne les attributs qui seront utilisés lors des prochaines actions.
5758	167E	Positionne le point indiqué par l'accumulateur graphique dans l'octet ATTRBYT.
5804	16AC	Les routines suivantes portent sur l'accumulateur graphique défini ci-dessus.
5804	16AC	Déplacement d'un point vers la droite avec indicateur C si atteinte d'un bord.
5829	16C5	Idem 16AC, sans indicateur C.
5848	16D8	Idem 16AC, mais vers la gauche.
5870	16EE	Idem 16C5, mais vers la gauche.
5898	170A	Idem 16AC, mais vers le bas.
5930	172A	Idem 16C5, mais vers le bas.
5948	173C	Idem 16AC, mais vers le haut.
5981	175D	Idem 16C5, mais vers le haut.
6153	1809	Routine de remplissage de figure (BOX-FILL).
6252	186C	Ecriture d'une CONFIG (1) en mode SCREEN 2. En entrée, A contient la CONFIG, HL, l'adresse dans la table et F3F2H, la couleur de CONFIG.
6343	18C7	Chargement du coefficient d'elliptisation pour l'instruction CIRCLE.
6351	18CF	Routine utilisée par l'instruction PAINT pour initialiser la couleur des bords.
6621	19DD	CASSETTE : attente, puis arrêt du moteur.
6633	19E9	CASSETTE : arrêt du moteur.
6641	19F1	CASSETTE : démarrage du moteur, puis écriture du HEADER.
6681	1A19	CASSETTE : écriture d'un octet.
6755	1A63	CASSETTE : lecture du HEADER.
6844	1ABC	CASSETTE : lecture d'un octet.

(1) CONFIG = CONFIGURATION (contenue dans TGC = Table du Générateur de Configuration).

Adresse Accès DEC	Adresse Accès HEX	Rôle
6982	1B46	Routine RST 18 : écriture sur l'écran ou sur l'imprimante suivant l'état de PRTFLG (F416H), le caractère à écrire est contenu dans A en entrée.
7011	1B63	Affichage du contenu de A à l'écran.
7103	1BBF	Table du générateur de caractères constituée de 256 * 8 octets. Fin en 23BEH.
9151	23BF	Point d'entrée principal de l'éditeur d'écran (texte Basic).
9164	23CC	Point d'entrée pour la saisie (INPUT) avec production du "?".
9209	23F9	Retour au Basic.
9273	2439 (2)	Table des caractères spéciaux avec l'adresse de traitement. Fin en 2459H.
9306	245A	Traitement du CR.
9412	24C4	Traitement de CTRL-C.
9445	24E5	Bascule de mode insertion.
9458	24F2	Insertion d'un blanc.
9552	2550	Effacement (DELETE) du caractère courant.
9569	2561	Effacement du caractère précédent.
9646	25AE	Effacement ligne.
9657	25B9	Effacement fin ligne.
9687	25D7	Ajoute à une ligne existante.
9720	25F8	Positionne sur le mot suivant.
9742	260E	Positionne sur le mot précédent.
9764	2624	Déplacement à droite.
9780	2634	Déplacement à gauche.

(2) Voir pour plus de détails : Table des valeurs et des pointeurs d'adresse de traitement des codes de contrôle.

Adresses contenues dans **Tables des routines d'exécution des mots-clés du Basic (RAM)** (10643 – 11231)

Adresse Accès DEC	*Adresse Accès HEX*	*Rôle*
9868	268C	A cette adresse, commencent les routines de traitement arithmétiques (utilisation de DAC et ARG) (1).
9868	268C	Soustraction double précision : DAC = DAC - ARG.
9882	269A	Addition double précision : DAC = DAC + ARG.
9978	26FA	Normalisation d'un résultat.
10044	273C	Routine d'arrondi.
10115	2783	Inversion du signe de DAC.
10135	2797	SHIFT DAC à gauche d'un chiffre décimal.
10147	27A3	SHIFT DAC à droite d'un chiffre décimal.
10230	27E6	Multiplication double précision : DAC = DAC * ARG.
10399	289F	Division double précision : DAC = DAC / ARG.
10643	2993	COSINUS : DAC=COS(DAC) : COS(DAC)= SIN(DAC+PI/2).
10668	29AC	SINUS : DAC=SIN(DAC).
10747	29FB	TANGENTE : DAC=TAN(DAC) : TAN(DAC)= SIN(DAC)/COS(DAC).
10772	2A14	ARCTANG : DAC=ATN(DAC).
10866	2A72	LOG : DAC=LOG(DAC).
11007	2AFF	RACINE : DAC=SQR(DAC).
11082	2B4A	EXPON : DAC=EXP(DAC).
11231	2BDF	RANDOM : DAC=RND.
11400	2C88	Evaluation des polynômes.
11505	2CF1	Table des constantes pour l'évaluation des fonctions transcendantes (SINUS, COS, TAN, LOG, PI, ATN). Cette table se termine en 2E70H.

(1) DAC = DIGITAL ACCUMULATOR, ARG = ARGUMENT ACCUMULATOR : Accumulateurs externes utilisés pour Z80 en RAM.

DAC = RA1 = REGISTRE AUXILIAIRE 1 (F776H-F7FDH)
ARG = RA2 = REGISTRE AUXILIAIRE 2 (F847H-F84EH)

Voir pour plus de précisions : Zones de travail des registres ou registres auxiliaires (chapitre RAM).

Adresse Accès DEC	*Adresse Accès HEX*	*Rôle*
11889	2E71	SIGN : A = SIGN(DAC).
11901	2E7D	ZERO : DAC=0.
11906	2E82	ABS : DAC=ABS(DAC).
11910	2E86	NEG : DAC=-DAC.
11927	2E97	SGN : A = SGN(DAC) pour les valeurs entières.
11953	2EB1	Pousse DAC dans la pile (SP).
11966	2EBE	Pousse un nombre simple précision (SIPR) pointé par HL dans DAC.
11969	2EC1	Pousse le contenu de BC et DE (SIPR) dans DAC.
11980	2ECC	Pousse un nombre simple précision (SIPR) de DAC vers BC et DE dans l'ordre CBED.
11990	2ED6	Pousse un nombre (SIPR) pointé par HL dans BC et DE dans l'ordre CBED.
11999	2EDF	Idem dans l'ordre EDCB.
12008	2EE8	Pousse un nombre (SIPR) de DAC vers la zone pointée par HL.
12011	2EEB	Pousse un nombre (SIPR) de la zone pointée par DE vers la zone pointée par HL.
12015	2EEF	Pousse un nombre pointé par DE vers la zone pointée par HL. ITD contient le type (F663H).
12037	2F05	Idem, mais de HL vers DAC.
12045	2F0D	Idem, mais de DAC vers HL.
12065	2F21	Compare deux nombres simple précision, le premier est dans BC et DE, le second est dans DAC. En sortie, A est 1 si DAC est >, A = -1 si DAC est < et A = 0 s'ils sont égaux.
12109	2F4D	Idem, mais avec deux entiers, le premier dans DE et le second dans HL.
12124	2F5C	Idem, mais avec deux nombres double précision, le premier dans ARG (F847H) et le second dans DAC (F7F6H).
12163	2F83	Idem 2F5C, mais avec résultat dans A inversé.
12170	2F8A	Convertit DAC en entier.
12185	2F99	Pousse le contenu de HL dans DAC et positionne ITD au format entier.

Voir également Table des routines arithmétiques (RAM) (12618 à 12858)

Adresse Accès DEC	Adresse Accès HEX	Rôle
12210	2FB2	Force DAC au format simple précision.
12218	2FBA	Convertit un nombre en double précision contenu dans DAC au format simple précision.
12232	2FC8	Convertit un nombre du format entier au format simple précision dans DAC.
12346	303A	Force DAC au format double précision.
12354	3042	Convertit un nombre contenu dans DAC de simple en double précision.
12376	3058	Force DAC au format chaîne de caractères.
12381	305D	Pousse INT(DAC) dans DE.
12478	30BE	FIX : FIX(DAC)=SGN(DAC)*INT(ABS(DAC)).
12495	30CF	INT : DAC=INT(DAC).
12618	314A	MULTIPLICATION format entier : DE=BC*DE.
12647	3167	SOUSTRACTION format entier : HL=DE-HL.
12658	3172	ADDITION format entier : HL=DE+HL.
12691	3193	MULTIPLICATION format entier : HL=DE*HL.
12774	31E6	DIVISION format entier : HL=DE/HL, reste dans DE.
12828	321C	NEGATION format entier : HL=-HL.
12843	322B	NEGATION
12858	323A	MODULO format entier : HL=DE-DE/HL*HL, DE=quotient.
12953	3299	Chargement d'une constante ASCII dans DAC. Cette routine évalue le nombre qui se trouve dans une chaîne de caractères pointée par HL, la stocke dans DAC et positionne ITD en fonction de type de constante. Cette routine s'arrête lorsqu'elle rencontre une valeur non numérique. Elle accepte les valeurs signées exprimées en entier, réel ou en notation scientifique. Le nombre est rendu dans la plus grande précision possible.
13322	340A	Routine de sortie du message "in" suivi du numéro de ligne.

Adresse Accès DEC	Adresse Accès HEX	Rôle
13349	3425	Routine de sortie de la valeur contenue dans DAC en fonction du format indiqué par les registres A, B et C.
13670	3566	Impression au format simple et double précision.
13734	35A6	Impression en format fixe suivi de chiffres décimaux.
13807	35EF	Traitement de la notation scientifique 'E'.
13926	3666	Pousse des 0 dans le BUFFER : HL pointe sur le buffer et A contient le nombre de 0 à pousser.
13934	366E	Pousse les 0 dans le BUFFER avec une virgule ou un point au milieu. En entrée, A contient la position du point décimal et C la position de la virgule. HL pointe sur le BUFFER.
13946	367A	Compte le nombre de virgule et fournit le résultat dans C.
13966	368E	Pousse les virgules et les points décimaux dans le BUFFER.
14003	36B3	Convertit un nombre simple ou double précision au format décimal.
14043	36DB	Convertit un entier au format décimal.
14106	371A	Convertit le nombre contenu dans DAC en binaire octal ou hexadécimal.
14162	3752	Prend la longueur et le chiffre le moins significatif de DAC pour un nombre réel.
14175	375F	Pousse un espace avant le nombre (pour les positifs).
14203	377B	Routine qui élimine les chiffres à droite du nombre dans DAC (avec arrondi). A contient, en entrée, le nombre de chiffres à éliminer.
14242	37A2	Compte le nombre de chiffres libres à droite du point décimal.
14260	37B4	Calcule le nombre de chiffres significatifs de la mantisse.
14280+ 14195	37C8+ 37D7	Exponentiation simple et double précision.

ADRESSES ROM

ADRESSES PRINCIPALES DU SYSTEME D'EXPLOITATION (MONITEUR SYSTEME)

Adresse Accès DEC	Adresse Accès HEX	Rôle
14399	383F	Exponentiation entière.
14426	385A	Routine d'exponentiation entière proprement dite.
14605	390D	HL=HL*DE.
14618	391A	Positionne l'indicateur de CARRY si ARG peut être converti en entier.
14638	392E	Table des "Routines d'exécution des mots-clés du Basic". Fin de la table en 3A3D (1).
14910	3A3E	Table des "Pointeurs des zones alphabétiques de TCT (fin en 3A71H) (1).
14962	3A72	Table de "Création des tokens" ou Table des mots-clés du Basic. Fin en 3D3AH (1).
15675	3D3B	Table de priorité des opérateurs. Fin en 3D46H (1).
15687 à 15696 15697 à 15733	3D47 à 3D50 3D51 à 3D75	Table de conversion de type de données (1) table des routines arithmétiques (1).
15734	3D76	Table des messages d'erreur. Chaque message se termine par un octet = 00. Fin en 3FD1H (1).
16338	3FD2	Message 'in'.
16343	3FD7	Message 'Ok', CR, LF.
16348	3FDC	Message 'Break'.
16354	3FE2	Recherche d'une entrée pour une instruction FOR dans le pointeur passé dans DE.

TCT : Table de Création des Tokens.

(1) Voir détails de la table dans ce même chapitre.

ADRESSES PRINCIPALES DE L'INTERPRETEUR BASIC

4000H*-7FFFH
16384D-32767D
Longueur = 16K octets

Adresse Accès DEC	Adresse Accès HEX	Rôle
16385	4001	Fonction INP.
16406	4016	Instruction OUT.
16412	401C	Instruction WAIT.
16441	4039	Traitement de la fin de programme.
16669	411D	Point d'entrée principal pour l'impression du message 'Ok' et le retour en mode saisie.
16692	4134	MAIN : retour au mode saisie.
17045	4295	Analyse le programme pour retrouver la ligne dont le numéro est contenu dans DE. En sortie, si l'indicateur de carry n'est pas positionné, la ligne n'est pas trouvée. Si le carry est positionné, alors la ligne est trouvée. L'indicateur de zéro permet de dire si la ligne est supérieure à toutes les lignes déjà existantes ou non.
17074	42B2	Point d'entrée principal du CRUNCHER. Le cruncher sert à convertir tous les mots réservés en CODE, les constantes en format interne et les lignes en format binaire (au moment de l'analyse d'une ligne Basic utilisateur).
17700	4524	Routine de traitement de l'instruction FOR.
17921	4601	Routine d'analyse d'une nouvelle instruction.
18022	4666	CHRGTR (RST10H).
18200	4718	Instruction DEFSTR.
18203	471B	Instruction DEFINT.
18205	471E	Instruction DEFSNG.
18209	4721	Instruction DEFDBL.

* *4000H= 16384D contient F3 (DISABLE INTERRUPT) (comme 0000H).* [annotation manuscrite : DI]

Remarque *: les points d'entrées des routines d'exécution des mots-clés du Basic sont contenus dans la table située entre 392EH et 3A3DH (voir détails de cette table dans ce même chapitre).*

Adresse Accès DEC	Adresse Accès HEX	Rôle
18281	4769	Lit un numéro de ligne à la position courante du texte.
18334	479E	Commande RUN.
18354	47B2	Instruction GOSUB.
18408	47E8	Instruction GOTO.
18465	4821	Instruction RETURN.
18525	485D	Instruction REM.
18543	486F	Instruction IF.
18560	4880	Instruction LET.
18660	48E4	Instruction ON GOTO.
18781	495D	Instruction RESUME.
18858	49AA	Instruction ERROR.
18869	49B5	Commande AUTO.
18917	49E5	Traitement de IF THEN ELSE.
18973	4A1D	Instruction LPRINT.
18980	4A24	Instruction PRINT.
19214	4B0E	Instruction LINE INPUT.
19359	4B9F	Instruction READ.
19551	4C5F	Routine d'évaluation de formule.
19911	4DC7	Routine EVAL : évaluation d'une expression.
20311	4F57	Traitement des opérateurs relationnels.
20437	4FD5	Fonction USR.
20509	501D	DEF FN.
20909	51AD	MID$ à gauche.
20937	51C9	Instruction WIDTH.
21033	5229	Instruction LLIST.
21038	522E	Instruction LIST.
21474	53E2	Commande DELETE.
21532	541C	Fonction PEEK.
21539	5423	Instruction POKE.
21608	5468	Instruction RENUM.

Adresse Accès DEC	Adresse Accès HEX	Rôle
21900	558C	SYNCHK (RST8).
21911	5597	GETYPR (RST28).
21928	55A8	Instruction CALL.
22428	579C	Début du GENGRP (Routines Graphiques Générales).
22428	579C	Scanning d'une coordonnée.
22501 +22506	57E5 +57EA	Instructions PRESET et PSET.
22531	5803	Fonction POINT.
22695	58A7	Instruction LINE.
22981	59C5	Instruction PAINT.
23313	5B11	Instruction CIRCLE.
23918	5D6E	Instruction DRAW.
24223	5E9F	Instruction DIM.
24753	60B1	Routine de traitement de PRINT USING.
25168	6250	Routine de transfert de blocs.
25205	6275	Traitement de l'erreur OUT OF MEMORY.
25222	6286	Commande NEW.
25249	62A1	CLEAR : effacement de toutes les variables.
25521	63B1	Routine de TRAPPING (ON - OFF - STOP).
25545	63C9	Instruction RESTORE.
25571	63E3	Instruction STOP.
25578	63EA	Instruction END.
25636	6424	Commande CONT.
25656 +25657	6438 +6439	Instruction TRON et TROFF.
25662	643E	Instruction SWAP.
25719	6477	Instruction ERASE.
25775	64AF	Instruction CLEAR.
25895	6527	Instruction NEXT.

Adresse Accès DEC	Adresse Accès HEX	Rôle
26056	65C8	Comparaison de deux chaînes de caractères. En entrée, HL doit pointer vers l'adresse de la première chaîne, BC doit pointer vers l'adresse de la seconde, D doit contenir la longueur de la première chaîne et E la longueur de la seconde. En sortie, le registre A contient FFH, 0 ou 1 suivant le résultat de la comparaison.
26101	65F5	Conversion d'un nombre en OCTAL (OCT$).
26106	65FA	Conversion d'un nombre en HEXADECIMAL (HEX$).
26111	65FF	Conversion d'un nombre en BINAIRE (BIN$).
26116	6604	Conversion d'un nombre contenu dans DAC en chaîne de caractères. En entrée, le nombre doit être dans DAC, ITD doit être positionné en fonction du type de variable. L'adresse de retour doit être poussée dans la PILE, ensuite, poussez HL puis BC, et enfin sautez (JP) à l'adresse 6604H. En sortie, le VARPTR de la chaîne se trouve dans DAC et ITD est positionné à 3.
26165	6635	Création d'un espace dans la zone des chaînes de caractères. En entrée, HL+1 pointe sur le premier caractère de la chaîne.
26202	665A	Création d'un VARPTR de chaîne de caractères. HL doit pointer sur le premier caractère de la chaîne, la chaîne doit se terminer par un 00H. En sortie, le VARPTR se trouve dans DAC avec ITD égal à 3.
26254	668E	STRING GARBAGE COLLECTION (regroupement des espaces pour les chaînes de caractères).
26503	6787	Concaténation de deux chaînes de caractères. En entrée, la PILE doit être préparée de la façon suivante : poussez l'adresse de retour, poussez BC puis HL. HL doit contenir l'adresse du VARPTR de la première chaîne et DAC doit contenir le VARPTR de la seconde, ITD doit valoir 3. Sautez à l'adresse 6787H (JP). En sortie, le VARPTR de la nouvelle chaîne se trouve dans DAC.

Adresse Accès DEC	Adresse Accès HEX	Rôle
26623	67FF	Fonction LEN. En entrée, le VARPTR de la chaîne doit être dans DAC, et ITD doit valoir 3. En sortie, la longueur est un entier dans DAC, et ITD vaut 2.
26635	680B	Fonction ASC. En entrée, le VARPTR de la chaîne se trouve dans DAC, ITD=3. En sortie, la valeur ASCII du premier caractère se trouve dans DAC, ITD=2.
26651	681B	Fonction CHR$. En entrée, DAC contient la valeur à convertir. En sortie, le VARPTR de la chaîne se trouve dans DAC, ITD=3.
26721	6861	Fonction LEFT$. En entrée, poussez l'adresse de retour dans la pile, ensuite poussez le VARPTR de la chaîne, chargez B avec le nombre de caractères à extraire et sautez à l'adresse 6861H. En sortie, DAC contient le VARPTR de la nouvelle chaîne.
26769	6891	Fonction RIGHT$. Fonctionne comme LEFT$.
26778	689A	Fonction MID$ à droite. En entrée, A contient la position du premier caractère à extraire, et E contient le nombre de caractères à extraire. Pour le reste, voir fonction LEFT$.
26811	68BB	Fonction VAL. En entrée, le VARPTR de la chaîne se trouve dans DAC avec ITD=3. En sortie, la valeur se trouve en double précision dans DAC avec ITD=8.
26859	68EB	Fonction INSTR.
27122	69F2	Fonction FRE.
27150	6A0E	Début des routines du Basic Disque.
27150	6A0E	Lecture du nom de fichier et du DEVICE.
27319	6AB7	Instruction OPEN.
27555	6BA3	Instruction SAVE.
27658	6C14	Instruction CLOSE.
27690 +27695	6C2A +6C2F	Instruction LFILES + FILES.
27701	6C35	Instruction PUT et GET.
27783	6C87	Instruction INPUT$.

Adresse Accès DEC	Adresse Accès HEX	Rôle
27907	6D03	Fonction LOC.
27924	6D14	Fonction LOF.
27941	6D25	Fonction EOF.
27961	6D39	Fonction FPOS.
28306	6E92	Instruction BSAVE.
28358	6EC6	Instruction BLOAD.
28437	6F15	Analyse du DEVICE* ou du DISQUE.
28534	6F76	Table des DEVICES : CAS-LPT-CRT-GRP.
28559	6F8F	Traitement du DEVICE.
28599	6FB7	Instruction CSAVE.
28735	703F	Instruction CLOAD.
28927	70FF	Message 'FOUND'.
28934	7106	Message 'SKIP'.
29480	7328	Envoi d'un CR suivi d'un LF (routine CRDO).
29623	73B7	MOTOR ON ou OFF.
29642	73CA	Instruction SOUND.
29669	73E5	Instruction PLAY.
30552 +30555	7758 +775B	Traitement de PUT et GET.
30566	7766	Instruction LOCATE.
30629	77A5	ON STOP.
30635	77AB	ON SPRITE.
30641	77B1	ON INTERVAL.
30655	77BF	ON STRIG.
30676	77D4	ON KEY.
30696	77E8	Routine ON KEY.

* *DEVICE : dispositif logique d'entrée ou de sortie.*

CAS : cassette
CRT : écran TV ou moniteur
LPT : imprimante
GRP : écran graphique

Adresse Accès DEC	Adresse Accès HEX	Rôle
30776	7838	Routine ON INTERVAL.
30828	786C	Instruction KEY (SET ou LIST).
30993	7911	Fonction TIME.
31003	791B	Suite de PLAY.
31040	7940	Fonction STICK.
31052	794C	Fonction STRIG.
31066	795A	Fonction PDL.
31081	7969	Fonction PAD.
31104	7980	Instruction COLOR.
31180	79CC	Instruction SCREEN.
31304	7A48	Instruction SPRITE.
31364	7A84	Fonction SPRITE.
31407	7AAF	Instruction PUT SPRITE.
31543	7B37	Instruction VDP.
31578	7B5A	Instruction BASE.
31651	7BA3	Table des valeurs par défaut pour BASE.
31691	7BCB	Fonction BASE.
31714	7BE2	Instruction VPOKE.
31733	7BF5	Fonction VPEEK.
31766	7C16	Suite de crochets pour les fonctions et instructions disques
31862	7C76	Suite de la routine d'initialisation.
32049	7D31	Ecriture des messages copyright.
32093	7D5D	Test de la mémoire.
32331	7E4B	Instruction MAX.
32472	7ED8	Table des messages d'initialisation.
32551	7F27	Petites routines qui vont s'installer en F380H (gestion des SLOTS).
32575	7F3F	Table des valeurs de la zone de communication. Elles sont copiées dans celle-ci à l'initialisation.
32767	7FFF	Fin de la ROM.

0001H-015CH
1D-348D
Longueur = 348 octets

Adresse vecteur DEC	Saut à adresse DEC	Adresse vecteur HEX	Saut à adresse HEX	Long vec-teur	Rôle
1-7	727	0001-0007	02D7	7	Routine principale d'initialisation (RESET).
8-11	9859	0008-000B	2683	4	RST 8 (SYNCHK) : voir contenu de la ROM en 558CH.
12-15	438	000C-001F	01B6	4	Gestion des slots.
16-19	9862	0010-0013	2686	4	RST 10 (CHRGTR) : voir contenu de la ROM en 4666H.
20-23	465	0014-0017	01D1	4	Gestion des slots.
24-27	6981	0018-001B	1B45	4	RST 18 (OUTDO) : voir contenu de la ROM en 1B45H.
28-31	535	001C-001F	0217	4	Gestion des slots.
32-35	5226	0020-0023	146A	4	RST 20 (DCOMPR) : voir contenu de la ROM en 146AH.
36-39	606	0024-0027	025E	4	Gestion des slots.
40-47	9865	0028-002F	2689	8	RST 28 (GETYPR) : voir contenu de la ROM en 2689H.
48-55	517	0030-0037	0205	8	RST 30 (CALLF) : voir contenu de la ROM en 0205H.
56-58	3132	0038-003A	0C3C	3	RST 38 (KEYINT) : voir contenu de la ROM en 0C3CH.
59-61	1181	003B-003D	049D	3	PSG : initialisation du PSG.
62-64	5021	003E-0040	139D	3	Initialisation des touches de fonction F1-F10.
65-67	1399	0041-0043	0577	3	VDP : extinction de l'écran.
68-70	1392	0044-0046	0570	3	VDP : allumage de l'écran.

Format vecteur 4 octets : C3 (JP) - XX (OMS) - XX (OPS) - 00 (Adresse)

Format vecteur 3 octets : C3 (JP) - XX (OMS) - XX (OPS) (Adresse)

PSG = Programmable Sound Generator (AY-3-8910)
VDP = Video Display Processor (TMS 9929A)

Adresse vecteur DEC	Saut à adresse DEC	Adresse vecteur HEX	Saut à adresse HEX	Long vec-teur	Rôle
71-73	1407	0047-0049	057F	3	VDP : écriture dans un registre (B=contenu, C=#reg).
74-76	2007	004A-004C	07D7	3	VDP : lecture VIDEORAM (HL=adresse→A=contenu).
77-79	1997	004D-004F	07CD	3	VDP : écriture VIDEORAM (HL=adresse, A=valeur).
80-82	2028	0050-0052	07EC	3	VDP : positionne une adresse en lecture (HL=adresse).
83-85	2015	0053-0055	07DF	3	VDP : positionne une adresse en écriture (HL=adresse).
86-88	2069	0056-0058	0815	3	VDP : écriture d'un caractère dans VIDEORAM un certain nombre de fois (HL=adresse, A=valeur, BC=compteur).
89-91	1807	0059-005B	070F	3	VDP : lecture d'un nombre de caractères dans la VIDEORAM (HL=adresse, DE=buffer de réception, BC=compteur).
92-94	1860	005C-005E	0744	3	VDP : écriture d'un buffer de caractères dans la VIDEORAM (HL=adresse, DE=buffer, BC=compteur).
95-97	2127	005F-0061	084F	3	VDP : initialisation du VDP en fonction du mode (0 à 3) (A=mode).
98-101	2039	0062-0065	07F7	4	VDP : positionne les couleurs de bord et de fond (voir contenu de la ROM en 07F7H).
102-104	5016	0066-0068	1398	3	Retour d'interruption avec vecteur crochet en 0FDD6H.
105-107	1704	0069-006B	06A8	3	VDP : positionnement des valeurs des tables.
108-110	1294	006C-006E	050E	3	VDP : positionnement en mode SCREEN 0.
111-113	1336	006F-0071	0538	3	VDP : positionnement en mode SCREEN 1.
114-116	1490	0072-0074	05D2	3	VDP : positionnement en mode SCREEN 2.
117-119	1567	0075-0077	061F	3	VDP : positionnement en mode SCREEN 3.

A = Accumulateur Z80 - B, C, DE, HL, registres Z80
VDP = Video Display Processor (TMS 9929A)

Adresse vecteur DEC	Saut à adresse DEC	Adresse vecteur HEX	Saut à adresse HEX	Long vec-teur	Rôle
120-122	1428	0078-007A	0594	3	VDP : force mode texte (SCREEN 0).
123-125	1460	007B-007D	05B4	3	VDP : force mode graphique 1 (SCREEN 1).
126-128	1538	007E-0080	0602	3	VDP : force mode graphique 2 (SCREEN 2).
129-131	1625	0081-0083	0659	3	VDP : force mode multicolore (SCREEN 3).
132-134	1764	0084-0086	06E4	3	VDP : si A contient le numéro du SPRITE en entrée, en sortie, HL contient son adresse dans la TGS.
135-137	1785	0087-0089	06F9	3	VDP : comme ci-dessus, mais HL contient son adresse dans la TAS.
138-140	1796	008A-008C	0704	3	VDP : fournit dans A la longueur d'un SPRITE (8 ou 32) en fonction du registre R1.
141-143	5392	008D-008F	1510	3	Ecriture d'un caractère en mode graphique (voir ROM).
144-146	1213	0090-0092	04BD	3	PSG : initialisation de la file (QUEUE).
147-149	4354	0093-0095	1102	3	PSG : écriture dans le PSG (E=valeur, A=numéro registre).
150-152	4366	0096-0098	110E	3	PSG : lecture d'un registre (A=#registre→A=contenu).
153-155	4548	0099-009B	11C4	3	PSG : tâche musicale (PLAY).
156-158	3434	009C-009E	0D6A	3	KBD : lecture d'une touche.
159-161	4299	009F-00A1	10CB	3	KBD : attente de pression d'une touche.
162-164	2236	00A2-00A4	08BC	3	CRT : affichage d'un caractère sur écran.

VDP = Video Display Processor (TMS 9929A)
PSG = Programmable Sound Generator (AY-3-8910)
TAS = Table des Attributs des Sprites (Lutins)
TGS = Table du Générateur des Sprites (Lutins)
KBD = KeyBoarD (clavier)
CRT = Cathode Ray Tube (écran TV ou moniteur)

Adresse vecteur DEC	Saut à adresse DEC	Adresse vecteur HEX	Saut à adresse HEX	Long vec-teur	Rôle
165-167	2141	00A5-00A7	085D	3	LPT : sortie d'un caractère sur imprimante.
168-170	2180	00A8-00AA	0884	3	LPT : test du mot d'état de l'imprimante.
171-173	2205	00AB-00AD	089D	3	LPT : conversion pour caractère non affichable.
174-176	9151	00AE-00F0	23BF	3	Point d'entrée principal (PINLIN) pour la saisie.
177-179	9173	00B1-00B3	23D5	3	Saisie d'entrée.
180-182	9164	00B4-00B6	23CC	3	Affichage d'un '?' et saisie (INPUT).
183-185	1135	00B7-00B9	046F	3	CTRL-C ?
186-188	1019	00BA-00BC	03FB	3	Processus de traitement du BREAK (CTRL-C).
189-191	4345	00BD-00BF	10F9	3	Idem 00BAH, avec HL=0000.
192-194	4371	00C0-00C2	1113	3	BEEP.
195-197	2120	00C3-00C5	0848	3	CLS.
198-200	2190	00C6-00C8	088E	3	Positionnement du curseur. H et L contiennent respectivement la valeur verticale et la valeur horizontale.
201-203	2854	00C9-00CB	0B26	3	Affichage des touches de fonction si OF3DEH ≠ 0.
204-206	2837	00CC-00CE	0B15	3	Effacement des touches de fonction.
207-209	2859	00CF-00D1	0B2B	3	Affichage des touches de fonction.
210-212	2107	00D2-00D4	083B	3	Retour à l'ancien mode SCREEN à la fin d'un programme ou à la suite d'une erreur.
213-215	4590	00D5-00D7	11EE	3	Lecture des manettes de jeux (A=0,1,2)→A=valeur.
216-218	4691	00D8-00DA	1253	3	Lecture des boutons de tir (A=0,1,2)→A=valeur.
219-221	4780	00DB-00DD	12AC	3	Lecture de la tablette graphique (PAD).
222-224	4723	00DE-00E0	1273	3	Lecture de la manette analogique (PDL).
225-227	6755	00E1-00E3	1A63	3	CAS : lecture du HEADER.

LPT = Line Printer (imprimante)
CAS = Cassettophone
HEADER = En tête de programme (avant bits d'informations)

Adresse vecteur DEC	Saut à adresse DEC	Adresse vecteur HEX	Saut à adresse HEX	Long vec-teur	Rôle
228-230	6844	00E4-00E6	1ABC	3	CAS : lecture d'un octet.
231-233	6633	00E7-00E9	19E9	3	CAS : arrêt du moteur.
234-236	6641	00EA-00EC	19F1	3	CAS : démarrage du moteur puis écriture du HEADER.
237-239	6681	00ED-00EF	1A19	3	CAS : écriture d'un octet.
240-242	6621	00F0-00F2	19DD	3	CAS : attente puis arrêt du moteur.
243-245	4996	00F3-00F5	1384	3	CAS : MOTOR ON ou OFF.
246-248	5355	00F6-00F8	14EB	3	Retourne le nombre d'octets qui reste dans la file.
249-251	5266	00F9-00FB	1492	3	Pose data dans la file.
252-254	5829	00FC-00FE	16C5	3	Déplace l'accumulateur graphique d'un point à droite.
255-257	5870	00FF-0101	16EE	3	Déplace l'accumulateur graphique d'un point à gauche.
258-260	5981	0102-0104	175D	3	Déplace l'accumulateur graphique d'un point en haut.
261-263	5948	0105-0107	173C	3	Déplace l'accumulateur graphique d'un point en haut.
264-266	5930	0108-010A	172A	3	Déplace l'accumulateur graphique d'un point en bas.
267-269	5898	010B-010D	170A	3	Déplace l'accumulateur graphique d'un point en bas.
270-272	5529	010E-0110	1599	3	Ajustage de X et Y (SCALXY) : voir ROM.
273-275	5599	0111-0113	15DF	3	(MAPXY) : voir ROM à l'adresse 15DFH.
276-278	5689	0114-0116	1639	3	Lecture de la valeur de l'accumulateur graphique.
279-281	5696	0117-0119	1640	3	Ecriture de la valeur de l'accumulateur graphique.
282-284	5750	011A-011C	1676	3	Positionne les attributs de l'accumulateur graphique qui seront utilisés lors des prochaines actions.
285-287	5703	011D-011F	1647	3	Lit les attributs de l'accumulateur graphique courant.
288-290	5758	0120-0122	167E	3	Positionne le point indiqué par l'accumulateur graphique dans l'octet ATTRBYT.
291-293	6153	0123-0125	1809	3	Routine de remplissage des rectangles (BOX FILL).

CAS = Cassettophone
HEADER = En tête de programme (avant bits d'informations)

Adresse vecteur DEC	Saut à adresse DEC	Adresse vecteur HEX	Saut à adresse HEX	Long vec-teur	Rôle
294-296	6343	0126-0128	18C7	3	Chargement du coefficient d'elliptisation du cercle.
297-299	6351	0129-012B	18CF	3	Initialisation de la couleur du bord (PAINT).
300-302	6372	012C-012E	18E4	3	Scrutation des points vers la droite (PAINT).
303-305	6522	012F-0131	197A	3	Scrutation des points vers la gauche (PAINT).
306-308	3901	0132-0134	0F3D	3	Eteint ou allume le témoin CAPS.
309-311	3962	0135-0137	0F7A	3	Positionne le BIT 7 du port C (SOUND).
312-314	5196	0138-013A	144C	3	PPI : lecture du port A du PPI (A=contenu).
315-317	5199	013B-013D	144F	3	PPI : écriture sur le port A (A=valeur lue).
318-320	5193	013E-0140	1449	3	VDP : lecture du registre d'état.
321-323	5202	0141-0143	1452	3	PPI : écriture du port C et lecture de B (CLAVIER).
324-326	5258	0144-0146	148A	3	Vecteur crochet vers OFFA7H pour extension.
327-329	5262	0147-0149	148E	3	Vecteur crochet vers OFFACH pour extension.
330-332	5215	014A-014C	145F	3	Test si OF864H #0 : présence de fichier.
333-335	7011	014D-014F	1B63	3	(OUTDLP) : sortie du contenu de A sur l'écran.
336-338	5232	0150-0152	1470	3	(GETVCP) A = voix (0,1,2) (PLAY).
339-341	5236	0153-0155	1474	3	(GETVC2) L = déplacement dans le buffer (PLAY).
342-344	1128	0156-0158	0468	3	KBD : effacement du tampon du clavier.
345-348	511	0159-015C	01FF	4	Gestion des slots.

PPI = Programmable Port Interface (8255)
VDP = Video Display Processor (TMS 9929A)
KBD = KeyBoarD (clavier).

TABLE 1
092FH-097FH
2351D-2431D
Longueur = 81 octets

TABLE 2
2439H-2459H
9273D-9305D
Longueur = 33 octets

Rôle

Chaque table regroupe, par bloc de trois octets, la valeur du code de contrôle (un octet) et le pointeur de l'adresse d'entrée de la routine de traitement de ce code de contrôle sous format OMS, OPS (deux octets).

Rappel : les codes de contrôle compris entre 01H et 7FH ou 1D et 127D permettent des fonctions d'édition à l'écran ou des fonctions sonores, exemple : BEEP.

- La table 1 comporte 27 groupes de 3 octets.
- La table 2 comporte 11 groupes de 3 octets.

Structure Table 1 (092FH-097FH)

Adresse HEX	*Adresse DEC*	*Contenu HEX*	*Code HEX*	*Code DEC*	*Routine HEX*	*Routine DEC*	*Rôle*
092F-0931	2351-2353	07-13-11	07	7	1113	4371	BEEP.
0932-0934	2354-2356	08-4C-0A	08	8	0A4C	2636	BACKSPACE.
0935-0937	2357-2359	09-71-0A	09	9	0A71	2673	Tabulation (8 espaces).
0938-093A	2360-2362	0A-08-09	0A	10	0908	2312	Interligne (LF).
093B-093D	2363-2365	0B-7F-0A	0B	11	0A7F	2687	Curseur début écran (HOME).
093E-0940	2366-2368	0C-7E-07	0C	12	077E	1918	Vide écran (CLS).
0941-0943	2369-2371	0D-81-0A	0D	13	0A81	2689	Retour Chariot (CR).
0944-0946	2372-2374	1B-89-09	1B	27	0989	2441	Echappement (ESC).
0947-0949	2375-2377	1C-5B-0A	1C	28	0A5B	2651	Curseur à droite (→).

Adresse HEX	*Adresse DEC*	*Contenu HEX*	*Code HEX*	*Code DEC*	*Routine HEX*	*Routine DEC*	*Rôle*
094A-094C	2378-2380	1D-4C-0A	1D	29	0A4C	2636	Curseur à gauche (←).
094D-094F	2381-2383	1E-57-0A	1E	30	0A57	2647	Curseur en haut (↑).
0950-0952	2384-2386	1F-61-0A	1F	31	0A61	2657	Curseur en bas (↓).
0953-0955	2387-2389	6A-1E-07	6A	106	077E	1918	Vide écran (CLS).
0956-0958	2390-2392	45-7E-07	45	69	077E	1918	Vide écran (CLS).
0959-095B	2393-2395	4B-EE-0A	4B	75	0AEE	2798	Effacement fin ligne.
095C-095E	2396-2398	4A-05-0B	4A	74	0B05	2821	Effacement fin page.
095F-0961	2399-2401	6C-EC-0A	6C	108	0AEC	2796	Effacement total ligne.
0962-0964	2402-2404	4C-B4-0A	4C	76	0AB4	2740	Insertion ligne.
0965-0967	2405-2407	4D-85-0A	4D	77	0A85	2693	Effacement ligne.
0968-096A	2408-2410	59-86-09	59	89	0986	2438	
096B-096D	2411-2413	41-57-0A	41	65	0A57	2647	Déplacement curseur ↑.
096E-0970	2414-2416	42-61-0A	42	66	0A61	2657	Déplacement curseur ↓.
0971-0973	2417-2419	43-44-0A	43	67	0A44	2628	Déplacement curseur →.
0974-0976	2420-2422	44-55-0A	44	68	0A55	2645	Déplacement curseur ←.
0977-0979	2423-2425	48-7F-0A	48	72	0A7F	2687	Curseur début écran (HOME).
097A-097C	2426-2428	78-80-09	78	120	0980	2432	
097D-097F	2429-2431	79-83-09	79	121	0983	2435	

Structure Table 2 (24394H-2459H)

Adresse HEX	*Adresse DEC*	*Contenu HEX*	*Code HEX*	*Code DEC*	*Routine HEX*	*Routine DEC*	*Rôle*
2439-243B	9273-9275	08-61-25	08	8	2561	9569	Effacement car précédent.
243C-243E	9276-9278	12-E5-24	12	18	24E5	9445	Choix mode insertion.
243F-2441	9279-9281	1B-FE-23	1B	27	23FE	9214	Echappement (ESC).
2442-2444	9282-9284	02-0E-26	02	2	260E	9742	Position sur mot précédent.
2445-2447	9285-9287	06-F8-25	06	6	25F8	9720	Position sur mot suivant.
2448-244A	9288-9290	0E-D7-25	0E	14	25D7	9687	Ajoute en fin de ligne.
244B-244D	9291-9293	0E-B9-25	05	5	25B9	9657	Efface fin de ligne.
244E-2450	9294-9296	03-C5-24	03	3	24C5	9413	Rupture en attente entrée.*
2451-2453	9297-9299	0D-5A-24	0D	13	245A	9306	Retour Chariot (CR).
2454-2456	9300-9302	15-AE-25	15	21	25AE	9646	Vide ligne syntaxique.
2457-2459	9303-9305	7F-50-25	7F	127	2550	9552	Effacement caractère courant (DEL).

* <CTRL><C>

13A9H-1448H
5033D-5192D
Longueur = 160 octets

Rôle

Cette table, d'une longueur de 160 octets, contient les valeurs par défaut (ou à l'initialisation) des dix touches de fonction (F1 à F10). A chaque touche, est affectée une zone mémoire de 16 octets. A l'initialisation, l'ensemble de la zone mémoire ROM 13A9H-1448H est recopié en ZONE DE COMMUNICATION RAM entre les adresses F87FH et F91EH (possibilité de modification par l'utilisateur).

Structure

Adresse HEX	*Adresse DEC*	*Touche / Long*	*CONTENU* { *1ère ligne : code ASCII (HEX)* / *2e ligne : contenu en clair* }
13A9-13B8	5033-5048	F1/16	63 6F 6C 6F 72 20 00 00 00 00 00 00 00 00 00 00 c o l o r
13B9-13C8	5049-5064	F2/16	61 75 74 6F 20 00 00 00 00 00 00 00 00 00 00 00 a u t o
13C9-13D8	5065-5080	F3/16	67 6F 74 6F 20 00 00 00 00 00 00 00 00 00 00 00 g o t o
13D9-13E8	5081-5096	F4/16	6C 69 73 74 20 00 00 00 00 00 00 00 00 00 00 00 l i s t
13E9-13F8	5097-5112	F5/16	72 75 6E 00 00 00 00 00 00 00 00 00 00 00 00 00 r u n (CR)*
13F9-1408	5113-5128	F6/16	63 6F 6C 6F 72 20 31 35 2C 34 2C 34 00 00 00 00 c o l o r 1 5 , 4 , 4 (CR)*
1409-1418	5129-5144	F7/16	63 6C 6F 61 64 22 00 00 00 00 00 00 00 00 00 00 c l o a d "
1419-1428	5145-5160	F8/16	63 6F 6E 74 0D 00 00 00 00 00 00 00 00 00 00 00 c o n t (CR)*
1429-1438	5161-5176	F9/16	6C 69 73 74 2E 00 1E 1E 00 00 00 00 00 00 00 00 l i s t .(CR)*(H)(H)
1439-1448	5177-5192	F10/16	0C 72 75 6E 00 00 00 00 00 00 00 00 00 00 00 00 (E) r u n (CR)*

* *(CR) = Carriage Return = retour chariot = ODH = 13D*
(E) = Effacement de l'écran (CLS)
(H) = Curseur en haut 1 position

392EH-3A3DH
14638D-14909D
Longueur = 272 octets

Rôle

- La **"Table des routines d'exécution des mots-clés du Basic"** contient les points d'entrée des routines d'exécution des TOKENS associés aux mots-clés du programme Basic (les tokens sont les codes sur un ou deux octets représentatifs des mots réservés du Basic frappés par le programmeur).
- La table est scrutée lors de la phase d'exécution du programme (après un RUN). Les points d'entrée (sur deux octets en format OMS,OPS → ADD DEC = OMS + 256 x OPS) sont classés du bas vers le haut de la table, suivant les numéros croissants des tokens.
- Sont classées dans la première partie (392EH ou 39DDH) les routines d'exécution des tokens sur un octet (de 81H à D8H), puis dans la deuxième partie (39DEH à 3A3DH) les routines d'exécution des tokens sur deux octets (FF-81H à FF-B0H).
- La table ne contient pas les routines d'exécution des mots-clés qui ne peuvent apparaître seuls (c'est-à-dire qui sont toujours associés à d'autres mots-clés). Exemple : TO, THEN, STEP, des mots-clés pour les opérateurs logiques ayant leurs propres routines : AND, OR, XOR,..., de certaines fonctions : VARPTR, USR, TAB.
- Sont en fait exclus de la table tous les mots-clés dont les numéros de token (un octet) sont compris entre D9H (TO) à FCH (\), zone où sont regroupés tous les tokens répondant aux restrictions mentionnées précedemment.

Description

* *Pour un groupe de deux octets*

Token DEC	Token HEX	Mot-clé	Position dans table*	Adresse routine HEX	Contenu HEX	Adresse DEC	Adresse HEX
129	81	END	0	63EA	EA-63	14638-14639	392E-392F
130	82	FOR	1	4524	24-45	14640-14641	3930-3931
131	83	NEXT	2	6527	27-65	14642-14643	3932-3933
132	84	DATA	3	485B	5B-48	14644-14645	3934-3935
133	85	INPUT	4	4B6C	6C-4B	14646-14647	3936-3937
134	86	DIM	5	5E9F	9F-5E	14648-14649	3938-3939
135	87	READ	6	4B9F	9F-4B	14650-14651	393A-393B
136	88	LET	7	4880	80-48	14652-14653	393C-393D
137	89	GOTO	8	47E8	E8-47	14654-14655	393E-393F
138	8A	RUN	9	479E	9E-47	14656-14657	3940-3941
139	8B	IF	10	49E5	E5-49	14658-14659	3942-3943
140	8C	RESTORE	11	63C9	C9-63	14660-14661	3944-3945
141	8D	GOSUB	12	47B2	B2-47	14662-14663	3946-3947
142	8E	RETURN	13	4821	21-48	14664-14665	3948-3949
143	8F	REM	14	485D	5D-48	14666-14667	394A-394B
144	90	STOP	15	63E3	E3-63	14668-14669	394C-394D
145	91	PRINT	16	4A24	24-4A	14670-14671	394E-394F
146	92	CLEAR	17	64AF	AF-64	14672-14673	3950-3951
147	93	LIST	18	522E	2E-52	14674-14675	3952-3953
148	94	NEW	19	6286	86-62	14676-14677	3954-3955
149	95	ON	20	48E4	E4-48	14678-14679	3956-3957
150	96	WAIT	21	401C	1C-40	14680-14681	3958-3959
151	97	DEF	22	501D	1D-50	14682-14683	395A-395B
152	98	POKE	23	5423	23-54	14684-14685	395C-395D
153	99	CONT	24	6424	24-64	14686-14687	395E-395F
154	9A	CSAVE	25	6FB7	B7-6F	14688-14689	3960-3961
155	9B	CLOAD	26	703F	3F-70	14690-14691	3962-3963
156	9C	OUT	27	4016	16-40	14692-14693	3964-3965

ADRESSES ROM

Description (suite)

Adresse HEX	Adresse DEC	Contenu HEX	Adresse routine HEX	Position dans table*	Mot-clé	Token HEX	Token DEC
3966-3967	14694-14695	1D-4A	4A1D	28	LPRINT	9D	157
3968-3969	14696-14697	29-52	5229	29	LLIST	9E	158
396A-396B	14698-14699	C3-00	00C3	30	CLS	9F	159
396C-396D	14700-14701	C9-51	51C9	31	WIDTH	A0	160
396E-396F	14702-14703	5D-48	485D	32	ELSE	A1	161
3970-3971	14704-14705	38-64	6438	33	TRON	A2	162
3972-3973	14706-14707	39-64	6439	34	TROFF	A3	163
3974-3975	14708-14709	3E-64	643E	35	SWAP	A4	164
3976-3977	14710-14711	77-64	6477	36	ERASE	A5	165
3978-3979	14712-14713	AA-49	49AA	37	ERROR	A6	166
397A-397B	14714-14715	5D-49	495D	38	RESUME	A7	167
397C-397D	14716-14717	E2-53	53E2	39	DELETE	A8	168
397E-397F	14718-14719	B5-49	49B5	40	AUTO	A9	169
3980-3981	14720-14721	68-54	5468	41	RENUM	AA	170
3982-3983	14722-14723	18-47	4718	42	DEFSTR	AB	171
3984-3985	14724-14725	1B-47	471B	43	DEFINT	AC	172
3986-3987	14726-14727	1E-47	471E	44	DEFSNG	AD	173
3988-3989	14728-14729	21-47	4721	45	DEFDBL	AE	174
398A-398B	14730-14731	0E-4B	4B0E	46	LINE	AF	175
398C-398D	14732-14733	B7-6A	6AB7	47	OPEN	B0	176
398E-398F	14734-14735	52-7C	7C52	48	FIELD	B1	177
3990-3991	14736-14737	5B-77	775B	49	GET	B2	178
3992-3993	14738-14739	58-77	7758	50	PUT	B3	179
3994-3995	14740-14741	14-6C	6C14	51	CLOSE	B4	180
3996-3997	14742-14743	5D-6B	6B5D	52	LOAD	B5	181
3998-3999	14744-14745	5E-6B	6B5E	53	MERGE	B6	182
399A-399B	14746-14747	2F-6C	6C2F	54	FILES	B7	183
399C-399D	14748-14749	48-7C	7C48	55	LSET	B8	184

Description (suite)

Adresse HEX	Adresse DEC	Contenu HEX	Adresse routine HEX	Position dans table*	Mot-clé	Token HEX	Token DEC
399E-399F	14750-14751	4D-7C	7C4D	56	RSET	B9	185
39A0-39A1	14752-14753	A3-6B	6BA3	57	SAVE	BA	186
39A2-39A3	14754-14755	2A-6C	6C2A	58	LFILES	BB	187
39A4-39A5	14756-14757	11-5B	5B11	59	CIRCLE	BC	188
39A6-39A7	14758-14759	80-79	7980	60	COLOR	BD	189
39A8-39A9	14760-14761	6E-5D	5D6E	61	DRAW	BE	190
39AA-39AB	14762-14763	C5-59	595C	62	PAINT	BF	191
39AC-39AD	14764-14765	C0-00	00C0	63	BEEP	C0	192
39AE-39AF	14766-14767	E5-73	73E5	64	PLAY	C1	193
39B0-39B1	14768-14769	EA-57	57EA	65	PSET	C2	194
39B2-39B3	14770-14771	E5-57	57E5	66	PRESET	C3	195
39B4-39B5	14772-14773	CA-73	73CA	67	SOUND	C4	196
39B6-39B7	14774-14775	CC-79	79CC	68	SCREEN	C5	197
39B8-39B9	14776-14777	E2-7B	7BE2	69	VPOKE	C6	198
39BA-39BB	14778-14779	48-7A	7A48	70	SPRITE	C7	199
39BC-39BD	14780-14781	37-7B	7B37	71	VDP	C8	200
39BE-39BF	14782-14783	5A-7B	7B5A	72	BASE	C9	201
39C0-39C1	14784-14785	A8-55	55A8	73	CALL	CA	202
39C2-39C3	14786-14787	11-79	7911	74	TIME	CB	203
39C4-39C5	14788-14789	6C-78	786C	75	KEY	CC	204
39C6-39C7	14790-14791	4B-7E	7E4B	76	MAX	CD	205
39C8-39C9	14792-14793	B7-73	73B7	77	MOTOR	CE	206
39CA-39CB	14794-14795	C6-6E	6EC6	78	BLOAD	CF	207
39CB-39CD	14796-14797	92-6E	6E92	79	BSAVE	D0	208
39CE-39CF	14798-14799	16-7C	7C16	80	DSKO$	D1	209
39D0-39D1	14800-14801	1B-7C	7C1B	81	SET	D2	210
39D2-39D3	14802-14803	20-7C	7C20	82	NAME	D3	211
39D4-39D5	14804-14805	25-7C	7C25	83	KILL	D4	212

Description (suite)

Adresse HEX	Adresse DEC	Contenu HEX	Adresse routine HEX	Position dans table*	Mot-clé	Token HEX	Token DEC
39D6-39D7	14806-14807	2A-7C	7C2A	84	IPL	D5	213
39D8-39D9	14808-14809	2F-7C	7C2F	85	COPY	D6	214
39DA-39DB	14810-14811	34-7C	7C34	86	CMD	D7	215
39DC-39DD	14812-14813	66-77	7766	87	LOCATE	D8	216
39DE-39DF	14814-14815	61-68	6861	88	LEFT$	FF-81	255-129
39E0-39E1	14816-14817	91-68	6891	89	RIGHT$	FF-82	255-130
39E2-39E3	14818-14819	9A-68	689A	90	MID$	FF-83	255-131
39E4-39E5	14820-14821	97-2E	2E97	91	SGN	FF-84	255-132
39E6-39E7	14822-14823	CF-30	30CF	92	INT	FF-85	255-133
39E8-39E9	14824-14825	82-2E	2E82	93	ABS	FF-86	255-134
39EA-39EB	14826-14827	FF-2A	2AFF	94	SQR	FF-87	255-135
39EC-39ED	14828-14829	DF-2B	2BDF	95	RND	FF-88	255-136
39EE-39EF	14830-14831	AC-29	29AC	96	SIN	FF-89	255-137
39F0-39F1	14832-14833	72-2A	2A72	97	LOG	FF-8A	255-138
39F2-39F3	14834-14835	4A-2B	2B4A	98	EXP	FF-8B	255-139
39F4-39F5	14836-14837	93-29	2993	99	COS	FF-8C	255-140
39F6-39F7	14838-14839	FB-29	29FB	100	TAN	FF-8D	255-141
39F8-39F9	14840-14841	14-2A	2A14	101	ATN	FF-8E	255-142
39FA-39FB	14842-14843	F2-69	69F2	102	FRE	FF-8F	255-143
39FC-39FD	14844-14845	01-40	4001	103	INP	FF-90	255-144
39FE-39FF	14846-14847	CC-4F	4FCC	104	POS	FF-91	255-145
3A00-3A01	14848-14849	FF-67	67FF	105	LEN	FF-92	255-146
3A02-3A03	14850-14851	04-66	6604	106	STR$	FF-93	255-147
3A04-3A05	14852-14853	BB-68	68BB	107	VAL	FF-94	255-148
3A06-3A07	14854-14855	0B-68	680B	108	ASC	FF-95	255-149
3A08-3A09	14856-14857	1B-68	681B	109	CHR$	FF-96	255-150
3A0A-3A0B	14858-14859	1C-54	541C	110	PEEK	FF-97	255-151
3A0C-3A0D	14860-14861	F5-7B	7BF5	111	VPEEK	FF-98	255-152

Description (suite)

Adresse HEX	Adresse DEC	Contenu HEX	Adresse routine HEX	Position dans table*	Mot-clé	Token HEX	Token DEC
3A0E-3A0F	14862-14863	48-68	6848	112	SPACE$	FF-99	255-153
3A10-3A11	14864-14865	F5-65	65F5	113	OCT$	FF-9A	255-154
3A12-3A13	14866-14867	FA-65	65FA	114	HEX$	FF-9B	255-155
3A14-3A15	14868-14869	C7-4F	4FC7	115	LPOS	FF-9C	255-156
3A16-3A17	14870-14871	FF-65	65FF	116	BIN$	FF-9D	255-157
3A18-3A19	14872-14873	8A-2F	2F8A	117	CINT	FF-9E	255-158
3A1A-3A1B	14874-14875	B2-2F	2FB2	118	CSNG	FF-9F	255-159
3A1C-3A1D	14876-14877	3A-30	303A	119	CDBL	FF-A0	255-160
3A1E-3A1F	14878-14879	BE-30	30BE	120	FIX	FF-A1	255-161
3A20-3A21	14880-14881	40-79	7940	121	STICK	FF-A2	255-162
3A22-3A23	14882-14883	4C-79	794C	122	STRIG	FF-A3	255-163
3A24-3A25	14884-14885	5A-79	795A	123	PDL	FF-A4	255-164
3A26-3A27	14886-14887	69-79	7969	124	PAD	FF-A5	255-165
3A28-3A29	14888-14889	39-7C	7C39	125	DSKF	FF-A6	255-166
3A2A-3A2B	14890-14891	39-6D	6D39	126	FPOS	FF-A7	255-167
3A2C-3A2D	14892-14893	66-7C	7C66	127	CVI	FF-A8	255-168
3A2E-3A2F	14894-14895	6B-7C	7C6B	128	CVS	FF-A9	255-169
3A30-3A31	14896-14897	70-7C	7C70	129	CVD	FF-AA	255-170
3A32-3A33	14898-14899	25-6D	6D25	130	EOF	FF-AB	255-171
3A34-3A35	14900-14901	03-6D	6D03	131	LOC	FF-AC	255-172
3A36-3A37	14902-14903	14-6D	6D14	132	LOF	FF-AD	255-173
3A38-3A39	14904-14905	57-7C	7C57	133	MKI$	FF-AE	255-174
3A3A-3A3B	14906-14907	5C-7C	7C5C	134	MKS$	FF-AF	255-175
3A3C-3A3D	14908-14909	61-7C	7C61	135	MKD$	FF-B0	255-176

* pour un groupe de deux octets.

3A3EH-3A71H
14910D-14961D
Longueur = 52 octets

Rôle

Cette table de 52 octets contient les 26 adresses de début des **"ZONES ALPHABETIQUES"** situées dans la **"TABLE DE CREATION DES TOKENS"** (392EH-3A3DH) ; cette dernière table contient 26 zones alphabétiques de A à Z (une par lettre de début des mots-clés) dont quatre sont vides (J, Q, Y, Z), c'est-à-dire ne contiennent aucun mot-clé débutant par les lettres J, Q, Y, Z. Ces zones vides ont quand même leurs pointeurs, mais leur contenu est limité à un octet 00H. Chaque pointeur est constitué de deux octets qui repèrent l'adresse dans la TCT sous format OMS, OPS.

Cette table est scrutée dès l'entrée d'une ligne Basic, après la frappe de la touche <ENTER>. L'interpréteur analyse chaque mot-clé introduit par le programmeur ; la détection de la première lettre du mot permet la scrutation de la table des pointeurs jusqu'à la position affectée à la lettre considérée. Le pointage adresse à ce moment la "ZONE ALPHABETIQUE" correspondante de la TCT et la création du token associé peut alors avoir lieu après analyse dans la **ZONE** des autres lettres du mot-clé.

Description

Adresse HEX	*Adresse DEC*	*Contenu HEX*	*Adresse HEX ZONE ALPHA*	*ZONE ALPHA*
3A3E-3A3F	14910-14911	72-3A	3A72	A
3A40-3A41	14912-14913	88-3A	3A88	B
3A42-3A43	14914-14915	9F-3A	3A9F	C
3A44-3A45	14916-14917	F3-3A	3AF3	D
3A46-3A47	14918-14919	2E-3B	3B2E	E
3A48-3A49	14920-14921	4F-3B	3B4F	F
3A4A-3A4B	14922-14923	69-3B	3B69	G
3A4C-3A4D	14924-14925	7B-3B	3B7B	H
3A4E-3A4F	14926-14927	80-3B	3B80	I
3A50-3A51	14928-14929	9F-3B	3B9F	J*
3A52-3A53	14930-14931	A0-3B	3BA0	K
3A54-3A55	14932-14933	A8-3B	3BA8	L
3A56-3A57	14934-14935	E8-3B	3BE8	M
3A58-3A59	14936-14937	09-3C	3C09	N
3A5A-3A5B	14938-14939	18-3C	3C18	O
3A5C-3A5D	14940-14941	2B-3C	3C2B	P

* *Zones vides*

Adresse HEX	*Adresse DEC*	*Contenu HEX*	*Adresse HEX ZONE ALPHA*	*ZONE ALPHA*
3A5E-3A5F	14942-14943	5D-3C	3C5D	Q*
3A60-3A61	14944-14945	5E-3C	3C5E	R
3A62-3A63	14946-14947	8E-3C	3C8E	S
3A64-3A65	14948-14949	DB-3C	3CDB	T
3A66-3A67	14950-14951	F6-3C	3CF6	U
3A68-3A69	14952-14953	FF-3C	3CFF	V
3A6A-3A6B	14954-14955	16-3D	3D16	W
3A6C-3A6D	14956-14957	20-3D	3D20	X
3A6E-3A6F	14958-14959	24-3D	3D24	Y*
3A70-3A71	14960-14961	25-3D	3D25	Z*

* *Zones vides*

3A72H-3D3AH 14962D-15674D Longueur = 713 octets

Rôle

- La "**TABLE DE CREATION DES TOKENS**" contient les "mots-clés" du Basic et leur équivalent codé que l'on appelle **TOKEN**. Le codage des mots-clés permet de sauvegarder de l'espace mémoire en affectant à un mot de plusieurs caractères un ou deux octets, uniquement pour le stockage en mémoire et la sauvegarde sur support magnétique. Cette façon de procéder est appelée "format compressé" ou "tokenisé" par rapport au format standard ASCII où tous les caractères des lignes d'instruction sont sauvegardés en code ASCII (un code pour un caractère).
- La table est scrutée dès que le programmeur frappe la touche <ENTER> en fin de ligne Basic. Les mots-clés sont classés par **ZONE ALPHABETIQUE**, chaque zone contient les mots-clés dont la première lettre est celle de l'indice de la zone (A, B, C, ... Y, Z). Les mots-clés ne sont pas rangés alphabétiquement dans chaque zone, mais sont classés par affinité (ainsi, AUTO, premier mot de la table, précède AND et ABS).
- Le mot-clé est stocké dans la table sous sa forme codée ASCII où ne figure pas la première lettre du code (la zone alphabétique indique quelle est la première lettre du mot) et où la dernière lettre du code est remplacée par son code ASCII+80H ou (code ASCII+128D) (valeurs comprises entre C1H et DAH ou 193D et 218D).
- Juste après le mot-clé, se trouve le code du mot clé ou TOKEN. Deux types de TOKENS peuvent apparaître, ceux codés sur un octet (valeurs comprises entre 81H et FCH), ceux codés sur deux octets (le premier octet est toujours FFH, le second octet est compris entre 81H et B0H). En fait, les tokens à deux octets ne sont pas représentés par deux octets dans la table, mais par un seul qui est l'index de déplacement par rapport à la valeur du token le plus bas -1 (80H = 128D).
- Ainsi le mot-clé "ASC" a un token = FFH-95H, alors que la valeur trouvée dans la table est 21H=33D, mais :

```
95H = 21H + 80H <-- token le plus bas -1
 ↑            \
 |             index de déplacement
2e octet du
token de ASC
```

Soit 2e octet du token = valeur stockée dans la TCT + 80H (ou 128D)
1er octet du token = FFH = 255D dans tous les cas.

- La fin d'une "**ZONE ALPHABETIQUE**" est repérée par un octet = 00H ; l'accès à la "**TABLE DE CREATION DES TOKENS**" se fait par l'intermédiaire d'une autre table située entre 3A3EH et 3A71H. Cette table contient, sous forme de bloc de deux octets (format OMS,OPS), les adresses de début des "**ZONES ALPHABETIQUES**" de la TCT, cette table est appelée "**TABLE DES POINTEURS DES ZONES ALPHABETIQUES DE LA TCT**".
- Si L est la longueur du mot-clé, la zone affectée à ce mot-clé est également de longueur L, suivant le format (L>2 caractères).

(L-2) octets	+	1 octet	+	1 octet
Code ASCII lettres du mot-clé (hors première et dernière)		Code ASCII+80H de la dernière lettre du mot-clé		Valeur du token (sur un octet) ou index de déplacement par rapport à 80H pour trouver 2e octet des tokens à 2 octets

- Si N est le nombre de mots-clés dans une "ZONE ALPHABETIQUE" donnée, la longueur LZ de la zone est :

$LZ = N \times L + 1$ ← octet 00H marqueur de fin de zone alphabétique

- Une "**ZONE ALPHABETIQUE**" vide (il y en a quatre dans la table pour les lettres J, Q, Y, Z où n'existe aucun mot-clé) est repérée par un octet 00 qui suit l'octet 00 de fin de ZONE ALPHABETIQUE précédente.
- A la fin de la table, entre 3D26H et 3D3AH (après la dernière zone alphabétique Z), se trouve la **zone des TOKENS pour les opérateurs** ; chaque opérateur est repéré dans la table par la valeur de son code ASCII+80H (ou 128D), il est suivi de la valeur du token associé.

Remarque : dans la table, les octets 00 de fin de zone alphabétique sont repérés par 00. Les deux octets de début de **zone** adressés par la table des pointeurs de TCT sont repérés par *.

ADRESSES ROM

TABLE DES MOTS-CLES DU BASIC ou TABLE DE CREATION DES TOKENS (TCT)

Description

Adresse HEX	Adresse DEC	Contenu HEX	Contenu HEX + caractères	Zone ALPHA	Long. en octets	Mot-clé	Token HEX	Token DEC
*3A72-3A75	*14962-14965	55-54-CF-A9	U-T+CF-A9	A	4	**AUTO**	A9	169
3A76-3A78	14966-14968	4E-C4-F6	N+C4-F6	A	3	**AND**	F6	246
3A79-3A7B	14969-14971	42-D3-06	B+D3-06	A	3	**ABS**	FF-86	255-134
3A7C-3A7E	14972-14974	54-CE-0E	T+CE-0E	A	3	**ATN**	FF-8E	FF-142
3A7F-3A81	14975-14977	53-C3-15	S+C3-15	A	3	**ASC**	FF-95	FF-149
3A82-3A87	14978-14983	54-54-52-A4-E9-[00]	T-T-R+A4-E9-[00]	A	5(+1)	**ATTR$**	E9	233
					22			
*3A88-3A8B	*14984-14987	41-53-C5-C9	A-S+C5-C9	B	4	**BASE**	C9	201
3A8C-3A90	14988-14992	53-41-56-C5-D0	S-A-V+C5-D0	B	5	**BSAVE**	D0	208
3A91-3A95	14993-14997	4C-4F-41-C4-CF	L-O-A+C4-CF	B	5	**BLOAD**	CF	207
3A96-3A99	14998-15001	45-45-D0-C0	E-E+D0-C0	B	4	**BEEP**	C0	192
3A9A-3A9E	15002-15006	49-4E-A4-1D-[00]	I-N+A4-AD-[00]	B	4(+1)	**BIN$**	FF-9D	255-157
					23			
*3A9F-3AA2	*15007-15010	41-4C-CC-CA	A-L+CC-CA	C	4	**CALL**	CA	202
3AA3-3AA7	15011-15015	4C-4F-53-C5-B4	L-O-S+C5-B4	C	5	**CLOSE**	B4	180
3AA8-3AAB	15016-15019	4F-50-D9-D6	O-P+D9-D6	C	4	**COPY**	D6	214
3AAC-3AAF	15020-15023	4F-4E-D4-99	O-N+D4-99	C	4	**CONT**	99	153
3AB0-3AB4	15024-15028	4C-45-41-D2-92	L-E-A+D2-92	C	5	**CLEAR**	92	146
3AB5-3AB9	15029-15033	4C-4F-41-C4-9B	L-O-A+C4-9B	C	5	**CLOAD**	9B	155
3ABA-3ABE	15034-15038	53-41-56-C5-9A	S-A-V+C5-9A	C	5	**CSAVE**	9A	154
3ABF-3AC4	15039-15044	53-52-4C-49-CE-E8	S-R-L-I+CE-E8	C	6	**CSRLIN**	E8	232
3AC5-3AC8	15045-15048	49-4E-D4-1E	I-N+D4-1E	C	4	**CINT**	FF-9E	255-158
3AC9-3ACC	15049-15052	53-4E-C7-1F	S-N+C7-1F	C	4	**CSNG**	FF-9F	255-159
3ACD-3AD0	15053-15056	44-42-CC-20	D-B+CC-20	C	4	**CDBL**	FF-A0	255-160
3AD1-3AD3	15057-15059	56-C9-28	V+C9-28	C	3	**CVI**	FF-A8	255-168
3AD4-3AD6	15060-15062	56-D3-29	V+D3-29	C	3	**CVS**	FF-A9	255-169
3AD7-3AD9	15063-15065	56-C4-2A	V+C4-2A	C	3	**CVD**	FF-AA	255-170

TABLE DES MOTS-CLES DU BASIC ou TABLE DE CREATION DES TOKENS (TCT)

Description (suite)

Adresse HEX	Adresse DEC	Contenu HEX	Contenu HEX + caractères	Zone ALPHA	Long. en octets	Mot-clé	Token HEX	Token DEC
3ADA-3ADC	15066-15068	4F-D3-0C	O+D3-0C	C	3	**COS**	FF-8C	255-140
3ADD-3AE0	15069-15072	48-52-A4-16	H-R+A4-16	C	4	**CHR$**	FF-96	250-150
3AE1-3AE6	15073-15078	49-52-43-4C-C5-BC	I-R-C-L+C5-B5	C	6	**CIRCLE**	BC	188
3AE7-3AEB	15079-15083	4F-4C-4F-D2-BD	O-L-O+D2-BD	C	5	**COLOR**	BD	189
3AEC-3AEE	15084-15086	4C-D3-9F	L+D3-9F	C	3	**CLS**	9F	159
3AEF-3AF2	15087-15090	4D-C4-D7-[00]	M+C4-D7-[00]	C	3(+1)	**CMD**	D7	215
					84			
*3AF3-3AF8	*15091-15096	45-4C-45-54-C5-A8	E-L-E-T+C5-A8	D	6	**DELETE**	A8	168
3AF9-3AFC	15097-15100	41-54-C1-84	A-T+C1-84	D	4	**DATA**	84	132
3AFD-3AFF	15101-15103	49-CD-86	I+CD-86	D	3	**DIM**	86	134
3B00-3B05	15104-15109	45-46-53-54-D2-AB	E-F-S-T+D2-AB	D	6	**DEFSTR**	AB	171
3B06-3B0B	15110-15115	45-46-49-4E-D4-AC	E-F-I-N+D4-AC	D	6	**DEFINT**	AC	172
3B0C-3B11	15116-15121	45-46-53-4E-C7-AD	E-F-S-N+C7-AD	D	6	**DEFSNG**	AD	173
3B12-3B17	15122-15127	45-46-44-42-CC-AE	E-F-D-B+CC-AE	D	6	**DEFDBL**	AE	174
3B18-3B1C	15128-15132	53-4B-4F-A4-D1	S-K-O+A4-D1	D	5	**DSKO$**	D1	209
3B1D-3B1F	15133-15135	45-C6-97	E+C6-97	D	3	**DEF**	97	151
3B20-3B24	15136-15140	53-4B-49-A4-EA	S-K-I+A4-EA	D	5	**DSKI$**	EA	234
3B25-3B28	15141-15144	53-4B-C6-26	S-K+C6-26	D	4	**DSKF**	FF-A6	255-166
3B29-3B2D	15145-15149	52-41-D7-BE-[00]	R-A+D7-BE-[00]	D	4(+1)	**DRAW**	BE	190
					59			

* Les octets soulignés sont les codes ASCII+80H des dernières lettres des mots-clés.

TABLE DES MOTS-CLES DU BASIC ou TABLE DE CREATION DES TOKENS (TCT)

Description (suite)

Adresse HEX	Adresse DEC	Contenu HEX	Contenu HEX + caractères	Zone ALPHA	Long. en octets	Mot-clé	Token HEX	Token DEC
*3B2E-3B31	*15150-15153	4C-53-C5-A1	L-S+C5-A1	E	4	ELSE	A1	161
3B32-3B34	15154-15156	4E-C4-81	N+C4-81	E	3	END	81	129
3B35-3B39	15157-15161	52-41-53-C5-A5	R-A-S+C5-A5	E	5	ERASE	A5	165
3B3A-3B3E	15162-15166	52-52-4F-D2-A6	R-R-O+D2-A6	E	5	ERROR	A6	166
3B3F-3B41	15167-15169	52-CC-E1	R+CC-E1	E	3	ERL	E1	225
3B42-3B44	15170-15172	52-D2-E2	R+D2-E2	E	3	ERR	E2	226
3B45-3B47	15173-15175	58-D0-0B	X+D0-0B	E	3	EXP	FF-8B	255-139
3B48-3B4A	15176-15178	4F-C6-2B	O+C6-2B	E	3	EOF	FF-AB	255-171
3B4B-3B4E	15179-15182	51-D6-F9- 00	Q+D6-F9- 00	E	3(+1)	EQV	F9	249
					33			
*3B4F-3B51	*15183-15185	4F-D2-82	O+D2-82	F	3	FOR	82	130
3B52-3B56	15186-15190	49-45-4C-C4-B1	I-E-L+C4-B1	F	5	FIELD	B1	177
3B57-3B5B	15191-15195	49-4C-45-D3-B7	I-L-E+D3-B7	F	5	FILES	B7	183
3B5C-3B5D	15196-15197	CE-DE	CE-DE	F	2	FN	DE	222
3B5E-3B60	15198-15200	52-C5-0F	R+C5-0F	F	3	FRE	FF-8F	255-143
3B61-3B63	15201-15203	49-D8-21	I-D8-21	F	3	FIX	FF-A1	255-161
3B64-3B68	15204-15208	50-4F-D3-27- 00	P-O+D3-27- 00	F	4(+1)	FPOS	FF-A7	255-167
					26			
*3B69-3B6C	*15209-15212	4F-54-CF-89	O-T+CF-89	G	4	GOTO	89	137
3B6D-3B71	15213-15217	4F-20-54-CF-89	O- -T+CF-89	G	5	GO TO	89	137
3B72-3B76	15218-15222	4F-53-55-C2-8D	O-S-U+C2-8D	G	5	GOSUB	8D	141
3B77-3B7A	15223-15226	45-D4-B2- 00	E+D4-B2- 00	G	3(+1)	GET	B2	178
					18			
*3B7B-3B7F	*15227-15231	45-58-A4-1B- 00	E-X+A4-1B	H	4(+1)	HEX$	FF-9B	255-155
					5			

* Les octets soulignés sont les codes ASCII+80H des dernières lettres des mots-clés.

TABLE DES MOTS-CLES DU BASIC ou TABLE DE CREATION DES TOKENS (TCT)

Description (suite)

Adresse HEX	Adresse DEC	Contenu HEX	Contenu HEX + caractères	Zone ALPHA	Long. en octets	Mot-clé	Token HEX	Token DEC
*3B80-3B84	*15232-15236	4E-50-55-D4-85	N-P-U+D4-85	I	5	INPUT	85	133
3B85-3B86	15237-15238	C6-8B	C6-8B	I	2	IF	8B	139
3B87-3B8B	15239-15242	4E-53-54-D2-E5	N-S-T+D2-E5	I	5	INSTR	E5	229
3B8C-3B8E	15244-15246	4E-D4-05	N+D4-05	I	3	INT	FF-85	255-133
3B8F-3B91	15247-15249	4E-D0-10	N+D0-10	I	3	INP	FF-90	255-144
3B92-3B94	15250-15252	4D-D0-FA	M+D0-FA	I	3	IMP	FA	250
3B95-3B9A	15253-15258	4E-4B-45-59-A4-EC	N-K-E-Y+A4-EC	I	6	INKEY$	EC	236
3B9B-3B9E	15259-15262	50-CC-D5- 00 -	P+CC-D5- 00	I	3(+1)	IPL	D5	213
					31			
*3B9F	*15263	00 zone vide	00 zone vide	J	(+1)			
					1			
*3BA0-3BA3	*15264-15267	49-4C-CC-D4	I-L+CC-D4	K	4	KILL	D4	212
3BA4-3BA7	15268-15271	45-D9-CC- 00	E+D9-CC- 00	K	3(+1)	KEY	CC	204
					8			
*3BA8-3BAD	*15272-15277	50-52-49-4E-D4-9D	P-R-I-N+D4-9D	L	6	LPRINT	9D	157
3BAE-3BB2	15278-15282	4C-49-53-D4-9E	L-I-S+D4-9E	L	5	LLIST	9E	158
3BB3-3BB6	15283-15286	50-4F-D3-1C	P-O+D3-1C	L	4	LPOS	FF-9C	255-156
3BB7-3BB9	15287-15289	45-D4-88	E+D4-88	L	3	LET	88	136
3BBA-3BBF	15290-15295	4F-43-41-54-C5-D8	O-C-A-T+C5-D8	L	6	LOCATE	D8	216
3BC0-3BC3	15296-15299	49-4E-C5-AF	I-N+C5-AF	L	4	LINE	AF	175
3BC4-3BC7	15300-15303	4F-41-C4-B5	O-A+C4-B5	L	4	LOAD	B5	181
3BC8-3BCB	15304-15307	53-45-D4-B8	S-E+D4-B8	L	4	LSET	B8	184
3BCC-3BCF	15308-15311	49-53-D4-93	I-S+D4-93	L	4	LIST	93	147
3BD0-3BD5	15312-15317	46-49-4C-45-D3-BB	F-I-L-E+D3-BB	L	6	LFILES	BB	187
3BD6-3BD8	15318-15320	4F-C7-0A	O+C7-0A	L	3	LOG	FF-8A	255-138
3BD9-3BDB	15321-15323	4F-C3-2C	O+C3-2C	L	3	LOC	FF-AC	255-172

TABLE DES MOTS-CLES DU BASIC ou
TABLE DE CREATION DES TOKENS (TCT)

Description (suite)

Adresse HEX	*Adresse DEC*	*Contenu HEX*	*Contenu HEX + caractères*	*Zone ALPHA*	*Long. en octets*	*Mot-clé*	*Token HEX*	*Token DEC*
3BDC-3BDE	15324-15326	45-CE-12	E+CE-12	L	3	LET	88	136
3BDF-3BE3	15327-15331	45-46-54-A4-01	E-F-T+A4-01	L	5	LEFT$	FF-81	255-129
3BE4-3BE7	15332-15335	4F-C6-2D-00	O+C6-ED-00	L	3(+1) / 64	LOF	FF-AD	255-173
*3BE8-3BEC	*15336-15340	4F-54-4F-D2-CE	O-T-O+D2-CE	M	5	MOTOR	CE	206
3BED-3BF1	15341-15345	45-52-47-C5-B6	E-R-G+C5-B6	M	5	MERGE	B6	182
3BF2-3BF4	15346-15348	4F-C4-FB	O+C4-FB	M	3	MOD	FB	251
3BF5-3BF8	15349-15352	4B-49-A4-2E	K-I+A4-2E	M	4	MKI$	FF-AE	255-174
3BF9-3BFC	15353-15356	4B-53-A4-2F	K-S+A4-2F	M	4	MKS$	FF-AF	255-175
3BFD-3C00	15357-15360	4B-44-A4-30	K-D+A4-30	M	4	MKD$	FF-B0	255-176
3C01-3C04	15361-15364	49-44-A4-03	I-D+A4-03	M	4	MID$	FF-83	255-131
3C05-3C08	15365-15368	41-D8-CD-00	A+D8-CD-00	M	3(+1) / 33	MAX	CD	205
*3C09-3C0C	*15369-15372	45-58-D4-83	E-X+D4-83	N	4	NEXT	83	131
3C0D-3C10	15373-15376	41-4D-C5-D3	A-M+C5-D3	N	4	NAME	D3	211
3C11-3C13	15377-15379	45-D7-94	E+D7-94	N	3	NEW	94	148
3C14-3C17	15380-15383	4F-D4-E0	O+D4-E0	N	3(+1) / 15	NOT	E0	224
*3C18-3C1B	*15384-15387	50-45-CE-B0	P-E+CE-B0	O	4	OPEN	B0	176
3C1C-3C1E	15388-15390	55-D4-9C	U+D4-9C	O	3	OUT	9C	156
3C1F-3C20	15391-15392	CE-95	CE-95	O	2	ON	95	149
3C21-3C22	15393-15394	D2-F7	D2-F7	O	2	OR	F7	247
3C23-3C26	15395-15398	43-54-A4-1A	C-T+A4-1A	O	4	OCT$	FF-9A	255-154
3C27-3C2A	15399-15402	46-C6-EB-00	F+C6-EB-00	O	3(+1) / 19	OFF	EB	235

* *Les octets soulignés sont les codes ASCII+80H des dernières lettres des mots-clés.*

TABLE DES MOTS-CLES DU BASIC ou
TABLE DE CREATION DES TOKENS (TCT)

Description (suite)

Adresse HEX	*Adresse DEC*	*Contenu HEX*	*Contenu HEX + caractères*	*Zone ALPHA*	*Long. en octets*	*Mot-clé*	*Token HEX*	*Token DEC*
*3C2B-3C2F	*15403-15407	52-49-4E-D4-91	R-I-N+D4-91	P	5	PRINT	91	145
3C30-3C32	15408-15410	55-D4-B3	U+D4-B3	P	3	PUT	B3	179
3C33-3C36	15411-15414	4F-4B-C5-98	O-K+C5-98	P	4	POKE	98	152
3C37-3C39	15415-15417	4F-D3-11	O+D3-11	P	3	POS	FF-91	255-145
3C3A-3C3D	15418-15421	45-45-CB-17	E-E+CB-17	P	4	PEEK	FF-97	255-151
3C3E-3C41	15422-15425	53-45-D4-C2	S-E+D4-C2	P	4	PSET	C2	194
3C42-3C47	15426-15431	52-45-53-45-D4-C3	R-E-S-E+D4-C3	P	6	PRESET	C3	195
3C48-3C4C	15432-15436	4F-49-4E-D4-ED	O-I-N+D4-ED	P	5	POINT	ED	237
3C4D-3C51	15437-15441	41-49-4E-D4-BF	A-I-N+D4-BF	P	5	PAINT	BF	191
3C52-3C54	15442-15444	44-CC-24	D+CC-24	P	3	PDL	FF-A4	255-164
3C55-3C57	15445-15447	41-C4-25	A+C4-25	P	3	PAD	FF-A5	255-165
3C58-3C5C	15448-15452	4C-41-D9-C1-00	L-A+D9-C1-00	P	4(+1) / 50	PLAY	C1	193
*3C5D	*15453	00 Zone vide	00 Zone vide	Q	(+1) / 1			
*3C5E-3C63	*15454-15459	45-54-55-52-CE-8E	E-T-U-R+CE-8E	R	6	RETURN	8E	142
3C64-3C67	15460-15463	45-41-C4-87	E-A+C4-87	R	4	READ	87	135
3C68-3C6A	15464-15466	55-CE-8A	U+CE-8A	R	3	RUN	8A	138
3C6B-3C71	15467-15473	45-53-54-4F-52-C5-8C	E-S-T-O-R+C5-8C	R	7	RESTORE	8C	140
3C72-3C74	15474-15476	45-CD-8F	E+CD-8F	R	3	REM	8F	143
3C75-3C7A	15477-15482	45-53-55-4D-C5-A7	E-S-U-M+C5-A7	R	6	RESUME	A7	167
3C7B-3C7E	15483-15486	53-45-D4-B9	S-E+D4-B9	R	4	RSET	B9	185
3C7F-3C84	15487-15492	49-47-48-54-A4-02	I-G-H-T+A4-02	R	6	RIGHT$	FF-82	255-130
3C85-3C87	15493-15495	4E-C4-08	N+C4-08	R	3	RND	FF-88	255-136
3C88-3C8D	15496-15501	45-4E-55-CD-AA-00	E-N-U+CD-AA-00	R	5(+1) / 48	RENUM	AA	170

Description (suite)

Adresse HEX	Adresse DEC	Contenu HEX	Contenu HEX + caractères	Zone ALPHA	Long. en octets	Mot-clé	Token HEX	Token DEC
*3C8E-3C93	*15502-15507	43-52-45-45-CE-C5	C-R-E-E+CE-C5	S	6	**SCREEN**	C5	197
3C94-3C99	15508-15513	50-52-49-54-C5-C7	P-R-I-T+C5-C7	S	6	**SPRITE**	C7	199
3C9A-3C9D	15514-15517	54-4F-D0-90	T-O+D0-90	S	4	**STOP**	90	144
3C9E-3CA1	15518-15521	57-41-D0-A4	W-A+D0-A4	S	4	**SWAP**	A4	164
3CA2-3CA4	15522-15524	45-D4-D2	E+D4-D2	S	3	**SET**	D2	210
3CA5-3CA8	15525-15528	41-56-C5-BA	A-V+C5-BA	S	4	**SAVE**	BA	186
3CA9-3CAC	15529-15532	50-43-A8-DF	P-C+A8-DF	S	4	**SPC(**	DF	223
3CAD-3CB0	15533-15536	54-45-D0-DC	T-E+D0-DC	S	4	**STEP**	DC	220
3CB1-3CB3	15537-15539	47-CE-04	6+CE-04	S	3	**SGN**	FF-84	255-192
3CB4-3CB6	15540-15542	51-D2-07	Q+D2-07	S	3	**SQR**	FF-87	255-135
3CB7-3CB9	15543-15545	49-CE-09	I+CE-09	S	3	**SIN**	FF-89	255-137
3CBA-3CBD	15546-15549	54-52-A4-13	T-R+A4-13	S	4	**STR$**	FF-93	255-147
3CBE-3CC4	15550-15556	54-52-49-4E-47-A4-E3	T-R-I-N-G+A4-E3	S	7	**STRING$**	E3	227
3CC5-3CCA	15557-15562	50-41-43-45-A4-19	P-A-C-E+A4-19	S	6	**SPACE$**	FF-99	255-153
3CCB-3CCF	15563-15567	4F-55-4E-C4-C4	O-U-N+C4-C4	S	5	**SOUND**	C4	196
3CD0-3CD4	15568-15572	54-49-43-CB-22	T-I-C+CB-22	S	5	**STICK**	FF-A2	255-162
3CD5-3CDA	15573-15578	54-52-49-C7-23-[00]	T-R-I+C7-23-[00]	S	5(+1) 77	**STRIG**	FF-A3	255-163
*3CDB-3CDE	*15579-15582	48-45-CE-DA	H-E+CE-DA	T	4	**THEN**	DA	218
3CDF-3CE2	15583-15586	52-4F-CE-A2	R-O+CE-A2	T	4	**TRON**	A2	162
3CE3-3CE7	15587-15591	52-4F-46-C6-A3	R-O-F+C6-A3	T	5	**TROFF**	A3	163
3CE8-3CEB	15592-15595	41-42-A8-DB	A-B+A8-DB	T	4	**TAB(**	DB	219
3CEC-3CED	15596-15597	CF-D9	CF-D9	T	2	**TO**	D9	217
3CEE-3CF1	15598-15601	49-4D-C5-CB	I-M+C5-CB	T	4	**TIME**	CB	203
3CF2-3CES	15602-15605	41-CE-OD-[00]	A+CE-OD-[00]	T	3(+1) 27	**TAN**	FF-8D	255-141

Description (suite)

Adresse HEX	Adresse DEC	Contenu HEX	Contenu HEX + caractères	Zone ALPHA	Long. en octets	Mot-clé	Token HEX	Token DEC
*3CF6-3CFA	*15606-15610	53-49-4E-C7-E4	S-I-N+C7-E4	U	5	**USING**	E4	228
3CFB-3CFE	15611-15614	53-D2-DD-[00]	S+D2-DD-[00]	U	3(+1) 9	**USR**	DD	221
*3CFF-3D01	*15615-15617	41-CC-14	A+CC-14	V	3	**VAL**	FF-94	255-148
3D02-3D07	15618-15623	41-52-50-54-D2-E7	A-R-P-T+D2-E7	V	6	**VARPTR**	E7	231
3D08-3D0A	15624-15626	44-D0-C8	D+D0-C8	V	3	**VDP**	C8	200
3D0B-3D0F	15627-15631	50-4F-4B-C5-C6	P-O-K+C5-C6	V	5	**VPOKE**	C6	198
3D10-3D15	15632-15637	50-45-45-CB-18-[00]	P-E-E+CB-18-[00]	V	5(+1) 23	**VPEEK**	FF-98	255-152
*3D16-3D1A	*15638-15642	49-44-54-C8-A0	I-D-T+C8-A0	W	5	**WIDTH**	A0	160
3D1B-3D1F	15643-15647	41-49-D4-96-[00]	A-I+D4-96-[00]	W	4(+1) 10	**WAIT**	96	150
*3D20-3D23	*15648-15651	4F-D2-F8-[00]	O+D2-F8-[00]	X	3(+1) 4	**XOR**	F8	248
*3D24	*15652	[00] Zone vide	[00] Zone vide	Y	(+1) 1			
*3D25	*15653	[00] Zone vide	[00] Zone vide	Z	(+1) 1			

* *Les octets soulignés sont les codes ASCII+80H des dernières lettres des mots-clés.*

Description (suite)

Adresse HEX	*Adresse DEC*	*Contenu HEX*	*Contenu HEX + caractères*	*Zone ALPHA*	*Long en octets*	*Mot-clé*	*Token HEX*	*Token DEC*
3D26-3D27	15654-15655	AB-F1	AB-F1	O	2	+	F1	241
3D28-3D29	15656-15657	AD-F2	AD-F2	P	2	-	F2	242
3D2A-3D2B	15658-15659	AA-F3	AA-F3	E	2	*	F3	243
3D2C-3D2D	15660-15661	AF-F4	AF-F4	R	2	/	F4	244
3D2E-3D2F	15662-15663	DE-F5	DE-F5	A	2	^	F5	245
3D30-3D31	15664-15665	DC-FC	DC-FC	T	2	\	FC	252
3D32-3D33	15666-15667	A7-E6	A7-E6	E	2	'(REM)	E6	230
3D34-3D35	15668-15669	BE-EE	BE-EE	U	2	>	EE	238
3D36-3D37	15670-15671	BD-EF	BD-EF	R	2	=	EF	239
3D38-3D3A	15672-15674	BC-F0-[00]	BC-F0-[00]	S	2(+1)	<	F0	240
					21			

Les octets soulignés sont les codes ASCII+80H des dernières lettres des mots-clés.

3D3BH-3D46H 15675D-15686D Longueur = 12 octets

Rôle

Détermine lors de calculs arithmétiques, mathématiques, logiques relationnels, les opérations à effectuer avant d'autres. Les rangs de priorité entre opérateurs définis dans cette table sont utilisés dans la phase d'analyse des instructions Basic, en conjugaison avec la pile opérationnelle qui permet le stockage des résultats intermédiaires.

Structure

Classement par adresse

Adresse HEX	*Adresse DEC*	*Contenu HEX*	*Contenu DEC*	*Opérateur*
3D3B	15675	79	121	+
3D3C	15676	79	121	-
3D3D	15677	7C	124	*
3D3E	15678	7C	124	/
3D3F	15679	7F	127	^
3D40	15680	50	80	AND
3D41	15681	46	70	OR
3D42	15682	3C	60	XOR
3D43	15683	32	50	EQV
3D44	15684	28	40	IMP
3D45	15685	7A	122	MOD
3D46	15686	7B	123	\

A chaque analyse d'expression, sont stockés dans la pile :

- l'opérateur ;
- la paire d'opérandes ;
- la valeur de priorité de l'opérateur précédent.

Quand un opérateur d'une plus grande valeur de priorité est rencontré, l'opération courante est effectuée et donne une valeur intermédiaire qui est stockée dans la pile.

TABLE DE PRIORITE DES OPERATEURS

Classement par ordre de priorité décroissante

Adresse HEX	*Adresse DEC*	*Contenu HEX*	*Contenu DEC*	*Opérateur*
3D3F	15679	7F	127	^
3D3D	15677	7C	124	*
3D3E	15678	7C	124	/
3D46	15686	7B	123	\
3D45	15685	7A	122	MOD
3D3B	15675	79	121	+
3D3C	15676	79	121	-
3D40	15680	50	80	AND
3D41	15681	46	70	OR
3D42	15682	3C	60	XOR
3D43	15683	32	50	EQV
3D44	15684	28	40	IMP

Opérateur relationnel (> < =)

Valeur de priorité = 64H = 100 D (par rapport aux autres opérateurs), mais non stockée dans la table.

Ordre de priorité calculé (entre opérateurs relationnels)

< =	06H	6D
< >	05H	5D
> =	04H	4D
<	03H	3D
=	02H	2D
>	01H	1D

TABLE DES ROUTINES DE CONVERSION DE TYPE DE DONNEES

3D47H-3D50H
15687D-15696D
Longueur = 10 octets

Rôle

Cette table, d'une longueur de dix octets, contient les pointeurs des points d'entrée des routines de conversion de type de données (conversion en double précision - DP, conversion en entier - E, conversion en chaîne de caractères - CC, conversion en simple précision - SP).

Ces pointeurs sont également trouvés dans la **"TABLE DES ROUTINES D'EXECUTION DES MOTS-CLES DU BASIC"** (392EH-3A3DH), car les routines de conversion en entier, simple précision, double précision correspondent aux mots-clés : CINT, CSNG, CDBL.

Structure

Adresse HEX	*Adresse DEC*	*Contenu HEX*	*Adresse routine HEX*	*Opération*	*ITD * final*	*Mot-clé*	*Adresse HEX dans TABLE EXECUTION*
3D47-3D48	15687-15688	3A-30	303A	Conversion en DP	8	**CBDL**	3A1C-3A1D
3D49-3D4A	15689-15690	00-00	néant	Néant	-		
3D4B-3D4C	15691-15692	8A-2F	2F8A	Conversion en E	2	**CINT**	3A18-3A19
3D4D-3D4E	15693-15694	58-30	3058	Conversion en CC	3		
3D4F-3D50	15695-15696	B2-2F	2FB2	Conversion en SP	4	**CSNG**	3A1A-3A1B

ITD = Indicateur de Type de Données (VAL TYPE).

3D51H-3D75H 15697D-15733D Longueur = 37 octets

Rôle

Cette table, d'une longueur de 37 octets, contient les pointeurs des points d'entrée des routines arithmétiques associés aux opérateurs arithmétiques : +, -, *, /, ^ et relationnels (comparaison) : >, <, =, en double précision - DP, simple précision - SP, et entier - E.

Le format de stockage est comme pour tous les pointeurs : OMS, OPS → adresse DEC = OMS + 256 * OPS.

Structure

Adresse HEX	*Adresse DEC*	*Contenu HEX*	*Adresse routine HEX*	*Opération*	*Opé-rateur*
3D51-3D52	15697-15698	9A-26	269A	Addition DP	+
3D53-3D54	15699-15700	8C-26	268C	Soustraction DP	-
3D55-3D56	15701-15702	E6-27	27E6	Multiplication DP	*
3D57-3D58	15703-15704	9F-28	289F	Division DP	/
3D59-3D5A	15705-15706	D7-37	37D7	Exponentiation DP	^
3D5B-3D5C	15707-15708	83-2F	2F83	Comparaison DP	>,<,=
3D5D-3D5E	15709-15710	4E-32	324E	Addition SP	+
3D5F-3D60	15711-15712	57-32	3257	Soustraction SP	-
3D61-3D62	15713-15714	5C-32	325C	Multiplication SP	*
3D63-3D64	15715-15716	67-32	3267	Division SP	/
3D65-3D66	15717-15718	C8-37	37C8	Exponentiation SP	^
3D67-3D68	15719-15720	21-2F	2F21	Comparaison SP	>,<,=
3D69-3D6A	15721-15722	72-31	3172	Addition E	+
3D6B-3D6C	15723-15724	67-31	3167	Soustraction E	-
3D6D-3D6E	15725-15726	93-31	3193	Multiplication E	*
3D6F-3D70	15727-15728	B8-4D	4DB8	Division E	/
3D71-3D72	15729-15730	3F-38	383F	Exponentiation E	^
3D73-3D74	15731-15732	4D-2F	2F4D	Comparaison E	>,<,=
3D75	15733	00	00	Fin de la table	

3D76H-3FD1H 15734D-16337D Longueur = 604 octets

Rôle

Les codes ASCII des messages d'erreurs (en clair) sont stockés les uns à la suite des autres, de façon jointive (avec séparateur 00H entre message). La position du message dans la table, à partir de son début (position 1 pour premier message), donne le code d'erreur n associé au message tel qu'il serait affiché par ERROR n.

Structure

[00] → marqueur de fin de message d'erreur.

Adresse HEX	*Adresse DEC*	*Code / Long*	*CONTENU* → 1ère ligne : code ASCII (HEX) / 2ème ligne : message d'erreur en clair	*Utilisation*
3D76-3D86	15734-15750	1/17	4E 45 58 54 20 77 69 74 68 6F 75 74 20 46 4F 52 [00] N E X T w i t h o u t F O R	BASIC
3D87-3D93	15751-15763	2/13	53 79 6E 74 61 78 20 65 72 72 6F 72 [00] S y n t a x e r r o r	BASIC
3D94-3DA8	15764-15784	3/21	52 45 54 55 52 4E 20 77 69 74 68 6F 75 74 20 47 4F 53 55 42 [00] R E T U R N w i t h o u t G O S U B	BASIC
3DA9-3DB4	15785-15796	4/12	4F 75 74 20 6F 66 20 44 41 54 41 [00] O u t o f D A T A	BASIC
3DB5-3DCA	15797-15818	5/22	49 6C 6C 65 67 61 6C 20 66 75 6E 63 74 69 6F 6E 20 63 61 6C 6C [00] I l l e g a l f u n c t i o n c a l l	BASIC
3DCB-3DD3	15819-15827	6/9	4F 76 65 72 66 6C 6F 77 [00] O v e r f l o w	BASIC

Adresse HEX	Adresse DEC	Code / Long	CONTENU → { 1ère ligne : code ASCII (HEX) 2ème ligne : message d'erreur en clair	Utilisation
3DD4-3DE1	15828-15841	7/14	4F 75 74 20 6F 66 20 6D 65 6D 6F 72 79 00 O u t o f m e m o r y	BASIC
3DE2-3DF7	15842-15863	8/22	55 6E 64 65 66 69 6E 65 64 20 6C 69 6E 65 20 6E 75 6D 62 65 72 00 U n d e f i n e d l i n e n u m b e r	BASIC
3DF8-3E0E	15864-15886	9/23	53 75 62 73 63 72 69 70 74 20 6F 75 74 20 6F 66 20 72 61 6E 67 65 00 S u b s c r i p t o u t o f r a n g e	BASIC
3E0F-3E22	15887-15906	10/20	52 65 64 69 6D 65 6E 73 69 6F 6E 65 64 20 61 72 72 61 79 00 R e d i m e n s i o n e d a r r a y	BASIC
3E23-3E33	15907-15923	11/17	44 69 76 69 73 69 6F 6E 20 62 79 20 7A 65 72 6F 00 D i v i s i o n b y z e r o	BASIC
3E34-3E42	15924-15938	12/15	49 6C 6C 65.67 61 6C 20 64 69 72 65 63 74 00 I l l e g a l d i r e c t	BASIC
3E43-3E50	15939-15952	13/14	54 79 70 65 20 6D 69 73 6D 61 74 63 68 00 T y p e m i s m a t c h	BASIC
3E51-3E64	15953-15972	14/20	4F 75 74 20 6F 66 20 73 74 72 69 6E 67 20 73 70 61 63 65 00 O u t o f s t r i n g s p a c e	BASIC
3E65-3E74	15973-15988	15/16	53 74 72 69 6E 67 20 74 6F 6F 20 6C 6F 6E 67 00 S t r i n g t o o l o n g	BASIC
3E75-3E8F	15989-16015	16/27	53 74 72 69 6E 67 20 66 6F 72 6D 75 6C 61 20 74 6F 6F 20 S t r i n g f o r m u l a t o o 63 6F 6D 70 6C 65 78 00 c o m p l e x	BASIC
3E90-3E9E	16016-16030	17/15	43 61 6E 27 74 20 43 4F 4E 54 49 4E 55 45 00 C a n ' t C O N T I N U E	BASIC

Adresse HEX	Adresse DEC	Code / Long	CONTENU → { 1ère ligne : code ASCII (HEX) 2ème ligne : message d'erreur en clair	Utilisation
3E9F-3EB6	16031-16054	18/24	55 6E 64 65 66 69 6E 65 64 20 73 73 65 72 20 66 75 6E 63 74 69 6F 6E 00 U n d e f i n e d u s e r f u n c t i o n	BASIC
3EB7-3EC7	16055-16071	19/17	44 65 76 69 63 65 20 49 2F 4F 20 65 72 72 6F 72 00 D e v i c e I / O e r r o r	BASIC
3EC8-3ED4	16072-16084	20/13	56 65 72 69 66 79 20 65 72 72 6F 72 00 V e r i f y e r r o r	BASIC
3ED5-3EDE	16085-16094	21/10	4E 6F 20 52 45 53 55 4D 45 00 N o R E S U M E	BASIC
3EDF-3EF3	16095-16115	22/21	52 45 53 55 4D 45 20 77 69 74 68 6F 75 74 20 65 72 72 6F 72 00 R E S U M E w i t h o u t e r r o r	BASIC
3EF4-3F05	16116-16133	23/18	55 6E 70 72 69 6E 74 61 62 6C 65 20 65 72 72 6F 72 00 U n p r i n t a b l e e r r o r	BASIC
3F06-3F15	16134-16149	24/16	4D 69 73 73 69 6E 67 20 6F 70 65 72 61 6E 64 00 M i s s i n g o p e r a n d	BASIC
3F16-3F2A	16150-16170	25/21	4C 69 6E 65 20 62 75 66 66 65 72 20 6F 76 65 72 66 6C 6F 77 00 L i n e b u f f e r o v e r f l o w	BASIC
3F2B*-3F39	16171-16185	50/15	46 49 45 4C 44 20 6F 76 65 72 66 6C 6F 77 00 F I E L D o v e r f l o w	BASIC DISQUE
3F3A-3F48	16186-16200	51/15	49 6E 74 65 72 6E 61 6C 20 65 72 72 6F 72 00 I n t e r n a l e r r o r	BASIC** DISQUE

* *La table, à partir de 3F2B, concerne les messages d'erreur du BASIC DISQUE. Le code d'erreur est égal à : (2p-p1) où p est la position dans la table par rapport au début de celle-ci, et p1 la position du message dans la table par rapport à 3F2B (début messages erreurs BASIC DISQUE).*

Adresse HEX	Adresse DEC	Code / Long	CONTENU → 1ère ligne : code ASCII (HEX)	2ème ligne : message d'erreur en clair	Utilisation
3F49-3F58	16201-16216	52/16	42 61 64 20 66 69 6C 65 20 6E 75 6D 62 65 72 (00)	B a d f i l e n u m b e r	BASIC ** DISQUE
3F59-3F67	16217-16231	53/15	46 69 6C 65 20 6E 6F 74 20 66 6F 75 6E 64 (00)	F i l e n o t f o u n d	BASIC DISQUE
3F68-3F79	16232-16249	54/18	46 69 6C 65 20 61 6C 72 65 61 64 79 20 6F 70 65 6E (00)	F i l e a l r e a d y o p e n	BASIC ** DISQUE
3F7A-3F88	16250-16264	55/15	49 6E 70 75 74 20 70 61 73 74 20 65 6E 64 (00)	I n p u t p a s t e n d	BASIC ** DISQUE
3F89-3F96	16265-16278	56/14	42 61 64 20 66 69 6C 65 20 6E 61 6D 65 (00)	B a d f i l e n a m e	BASIC ** DISQUE
3F97-3FAF	16279-16303	57/25	44 69 72 65 63 74 20 73 74 61 74 65 6D 65 6E 74 20 69 6E 20 66 69 6C 65 (00)	D i r e c t s t a t e m e n t i n f i l e	BASIC ** DISQUE
3FB0-3FC3	16304-16323	58/20	53 65 71 75 65 6E 74 69 61 6C 20 49 2F 4F 20 6F 6E 6C 79 (00)	S e q u e n t i a l I / O o n l y	BASIC DISQUE
3FC4-3FD1	16324-16337	59/14	46 69 6C 65 20 6E 6F 74 20 4F 50 45 4E (00)	F i l e n o t O P E N	BASIC ** DISQUE

Les messages d'erreur dont les codes sont ⩾ 60 (compris entre 60 et 72) correspondent au MSX-DOS et sont gérés en RAM par celui-ci ; ils ne figurent donc pas dans cette table réservée au BASIC + BASIC DISQUE.

*** Peut être également rencontré avec BASIC (version cassette).*

Classement par ordre alphabétique des mots-clés

Les mots-clés, suivis par un astérisque, sont ceux codés sur deux octets (tokens). Le premier octet vaut toujours, dans ce cas, FFH=255D, le second est celui qui est mentionné dans la colonne TOKEN.

Mot-clé	Adresse DEC	Adresse HEX	Token DEC	Token HEX
ABS *	11906	2E82	134	86
ASC *	26635	680B	149	95
ATN *	10772	2A14	142	8E
AUTO	18869	49B5	169	A9
BASE	31578	7B5A	201	C9
BEEP	192	00C0	192	C0
BIN$ *	26111	65FF	157	9D
BLOAD	28358	6EC6	207	CF
BSAVE	28306	6E92	208	D0
CALL	21928	55A8	202	CA
CDBL *	12346	303A	160	A0
CHR$ *	26651	681B	150	96
CINT *	12170	2F8A	158	9E
CIRCLE	23313	5B11	188	BC
CLEAR	25775	64AF	146	92
CLOAD	28735	703F	155	9B
CLOSE	27668	6C14	180	B4
CLS	195	00C3	159	9F
CMD	31796	7C34	215	D7
COLOR	34104	7980	189	BD
CONT	25636	6424	153	99
COPY	31791	7C2F	214	D6
COS *	10643	2993	140	8C
CSAVE	28599	6FB7	154	9A
CSNG *	12210	2FB2	159	9F
CVD *	31856	7C70	170	AA
CVI *	31846	7C66	168	A8
CVS *	31851	7C6B	169	A9
DATA	18523	485B	132	84
DEF	20509	501D	151	97
DEFDBL	18209	4721	174	AE
DEFINT	18203	471B	172	AC
DEFSNG	18206	471E	173	AD
DEFSTR	18200	4718	171	AB
DELETE	21474	53E2	168	A8
DIM	24223	5E9F	134	86
DRAW	23918	5D6E	190	BE
DSKF *	31801	7C39	166	A6
DSKO$	31766	7C16	209	D1

Classement par ordre alphabétique des mots-clés (suite)

Mot-clé	Adresse DEC	Adresse HEX	Token DEC	Token HEX
ELSE	18525	485D	161	A1
END	25578	63EA	129	81
EOF *	27941	6D25	171	AB
ERASE	25719	6477	165	A5
ERROR	18858	49AA	166	A6
EXP *	11082	2B4A	139	8B
FIELD	31826	7C52	177	B1
FILES	27695	6C2F	183	B7
FIX *	12472	30BE	161	A1
FOR	17700	4524	130	82
FPOS *	27961	6D39	167	A7
FRE *	27122	69F2	143	8F
GET	30555	775B	178	B2
GOSUB	18354	47B2	141	8D
GOTO	18408	47E8	137	89
HEX$ *	26106	65FA	155	9B
IF	18917	49E5	139	8B
INP *	16385	4001	144	90
INPUT	19308	4B6C	133	85
INT *	12495	30CF	133	85
IPL	31786	7C2A	213	D5
KEY	30828	786C	204	CC
KILL	31781	7C25	212	D4
LEFT$ *	26721	6861	129	81
LEN *	26623	67FF	146	92
LET	18560	4880	136	88
LFILES	27690	6C2A	187	BB
LINE	19214	4B0E	175	AF
LIST	21038	522E	147	93
LLIST	21033	5229	158	9E
LOAD	27485	6B5D	181	B5
LOC *	27907	6D03	172	AC
LOCATE	30566	7766	216	D8
LOF *	27924	6D14	173	AD
LOG *	10866	2A72	138	8A
LPOS *	20423	4FC7	156	9C
LPRINT	18973	4A1D	157	9D
LSET	31816	7C48	184	B8
MAX	32331	7E4B	205	CD
MERGE	27486	6B5E	182	B6
MID$ *	26778	689A	131	83
MKD$ *	31841	7C61	176	B0
MKI$ *	31831	7C57	174	AE
MKS$ *	31836	7C5C	175	AF

Classement par ordre alphabétique des mots-clés (suite)

Mot-clé	Adresse DEC	Adresse HEX	Token DEC	Token HEX
MOTOR	29623	73B7	206	CE
NAME	31776	7C20	211	D3
NEW	25222	6286	148	94
NEXT	25895	6527	131	83
OCT$ *	26101	65F5	154	9A
ON	18660	48E4	149	95
OPEN	27319	6AB7	176	B0
OUT	16406	4016	156	9C
PAD *	31081	7969	165	A5
PAINT	22981	59C5	191	BF
PDL *	31066	795A	164	A4
PEEK *	21532	541C	151	97
PLAY	29669	73E5	193	C1
POKE	21539	5423	152	98
POS *	20428	4FCC	145	91
PRESET	22501	57E5	195	C3
PRINT	18980	4A24	145	91
PSET	22506	57EA	194	C2
PUT	30552	7758	179	B3
READ	19359	4B9F	135	87
REM	18525	485D	143	8F
RENUM	21608	5468	170	AA
RESTORE	25545	63C9	140	8C
RESUME	18781	495D	167	A7
RETURN	18465	4821	142	8E
RIGHT$*	26769	6891.	130	82
RND *	11231	2BDF	136	88
RSET	31821	7C4D	185	B9
RUN	18334	479E	138	8A
SAVE	27555	6BA3	186	BA
SCREEN	31180	79CC	197	C5
SET	31771	7C1B	210	D2
SGN *	11927	2E97	132	84
SIN *	10668	29AC	137	89
SOUND	25546	73CA	196	C4
SPACE$ *	26696	6848	153	99
SPRITE	31304	7A48	199	C7
SQR *	11007	2AFF	137	87
STRIG *	31052	794C	162	A3
SWAP	25662	643E	164	A4
TAN *	10747	29FB	141	8D
TIME	30993	7911	203	CB
TROFF	25657	6439	163	A3
TRON	25656	6438	162	A2

Classement par ordre alphabétique des mots-clés (suite)

Mot-clé	Adresse DEC	Adresse HEX	Token DEC	Token HEX
VAL *	26811	68BB	148	94
VDP	31543	7B37	200	C8
VPEEK *	31733	7BF5	152	98
VPOKE	31714	7BE2	198	C6
WAIT	16412	401C	150	96
WIDTH	20937	51C9	150	96

Les mots-clés n'apparaissant pas dans cette liste sont ceux qui ne peuvent se produire seuls, c'est-à-dire qui sont toujours associés avec un autre mot-clé :

TO associé à FOR,
USING associé à PRINT,
THEN associé à IF,
ELSE associé à IF,... etc,

les opérateurs logiques (AND, OR, XOR, EQV, IMP) et les fonctions (VARPTR, USR, TAB).

Classement par numéro de token

Mot-clé	Adresse routine HEX	Adresse routine DEC	Token HEX	Token DEC
END	63EA	25578	81	129
FOR	4524	17700	82	130
NEXT	6527	25895	83	131
DATA	485B	18523	84	132
INPUT	4B6C	19308	85	133
DIM	5E9F	24223	86	134
READ	4B9F	19359	87	135
LET	4880	18560	88	136
GOTO	47E8	18408	89	137
RUN	479E	18334	8A	138
IF	49E5	18917	8B	139
RESTORE	63C9	25545	8C	140
GOSUB	47B2	18354	8D	141
RETURN	4821	18465	8E	142
REM	485D	18525	8F	143
STOP	63E3	22571	90	144
PRINT	4A24	18980	91	145
CLEAR	64AF	25775	92	146
LIST	522E	21038	93	147
NEW	6286	25222	94	148
ON	48E4	18660	95	149
WAIT	401C	16412	96	150
DEF	501D	20509	97	151
POKE	5423	21539	98	152
CONT	6424	25636	99	153
CSAVE	6FB7	28599	9A	154
CLOAD	703F	28735	9B	155
OUT	4016	16406	9C	156
LPRINT	4A1D	18973	9D	157
LLIST	5229	21033	9E	158
CLS	00C3	195	9F	159
WIDTH	51C9	20937	A0	160
ELSE	485D	18525	A1	161
TRON	6438	25656	A2	162
TROFF	6439	25657	A3	163
SWAP	643E	25662	A4	164
ERASE	6477	25719	A5	165
ERROR	49AA	18858	A6	166
RESUME	495D	18781	A7	167
DELETE	53E2	21474	A8	168
AUTO	49B5	18869	A9	169
RENUM	5468	21608	AA	170
DEFSTR	4718	18200	AB	171
DEFINT	471B	18203	AC	172
DEFSNG	471E	18206	AD	173

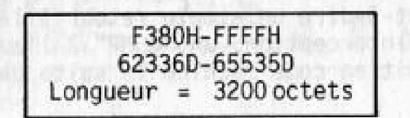
Classement par numéro de token (suite)

Mot-clé	Adresse routine HEX	Adresse routine DEC	Token HEX	Token DEC
DEFDBL	4721	18209	AE	174
LINE	4B0E	19214	AF	175
OPEN	6AB7	27319	B0	176
FIELD	7C52	31826	B1	177
GET	775B	30555	B2	178
PUT	7758	30552	B3	179
CLOSE	6C14	27668	B4	180
LOAD	6B5D	27485	B5	181
MERGE	6B5E	27486	B6	182
FILES	6C2F	27695	B7	183
LSET	7C48	31816	B8	184
RSET	7C4D	31821	B9	185
SAVE	6BA3	27555	BA	186
LFILES	6C2A	27690	BB	187
CIRCLE	5B11	23313	BC	188
COLOR	7980	34104	BD	189
DRAW	5D6E	23918	BE	190
PAINT	595C	22981	BF	191
BEEP	00C0	192	C0	192
PLAY	73E5	29669	C1	193
PSET	57EA	22506	C2	194
PRESET	57E5	22501	C3	195
SOUND	73CA	25546	C4	196
SCREEN	79CC	31180	C5	197
VPOKE	7BE2	31714	C6	198
SPRITE	7A48	31304	C7	199
VDP	7B37	31543	C8	200
BASE	7B5A	31578	C9	201
CALL	55A8	21928	CA	202
TIME	7911	30993	CB	203
KEY	786C	30828	CC	204
MAX	7E4B	32331	CD	205
MOTOR	73B7	29623	CE	206
BLOAD	6EC6	28358	CF	207
BSAVE	6E92	28306	D0	208
DSKO$	7C16	31766	D1	209
SET	7C1B	31771	D2	210
NAME	7C20	31776	D3	211
KILL	7C25	31781	D4	212
IPL	7C2A	31786	D5	213
COPY	7C2F	31791	D6	214
CMD	7C34	31796	D7	215
LOCATE	7766	30566	D8	216
LEFT$	6861	26721	FF-81	255-129
RIGHT$	6891	26769	FF-82	255-130
MID$	689A	26778	FF-83	255-131

Classement par numéro de token (suite)

Mot-clé	Adresse routine HEX	Adresse routine DEC	Token HEX	Token DEC
SGN	2E97	11927	FF-84	255-132
INT	30CF	12495	FF-85	255-133
ABS	2E82	11906	FF-86	255-134
SQR	2AFF	11007	FF-87	255-135
RND	2BDF	11231	FF-88	255-136
SIN	29AC	10668	FF-89	255-137
LOG	2A72	10866	FF-8A	255-138
EXP	2B4A	11082	FF-8B	255-139
COS	2993	10643	FF-8C	255-140
TAN	29FB	10747	FF-8D	255-141
ATN	2A14	10772	FF-8E	255-142
FRE	69F2	27122	FF-8F	255-143
INP	4001	16385	FF-90	255-144
POS	4FCC	20428	FF-91	255-145
LEN	67FF	26623	FF-92	255-146
STR$	6604	26116	FF-93	255-147
VAL	68BB	26811	FF-94	255-148
ASC	680B	26635	FF-95	255-149
CHR$	681B	26651	FF-96	255-150
PEEK	541C	21532	FF-97	255-151
VPEEK	78F5	31733	FF-98	255-152
SPACE$	6848	26696	FF-99	255-153
OCT$	65F5	26101	FF-9A	255-154
HEX$	65FA	26106	FF-9B	255-155
LPOS	4FC7	20423	FF-9C	255-156
BIN$	65FF	26111	FF-9D	255-157
CINT	2F8A	12170	FF-9E	255-158
CSNG	2FB2	12210	FF-9F	255-159
CDBL	303A	12346	FF-A0	255-160
FIX	30BE	12472	FF-A1	255-161
STICK	7940	31040	FF-A2	255-162
STRIG	794C	31052	FF-A3	255-163
PDL	795A	31066	FF-A4	255-164
PAD	7969	31081	FF-A5	255-165
DSKF	7C39	31801	FF-A6	255-166
FPOS	6D39	27961	FF-A7	255-167
CVI	7C66	31846	FF-A8	255-168
CVS	7C6B	31851	FF-A9	255-169
CVD	7C70	31856	FF-AA	255-170
EOF	6D25	27941	FF-AB	255-171
LOC	6D03	27907	FF-AC	255-172
LOF	6D14	27924	FF-AD	255-173
MKI$	7C57	31831	FF-AE	255-174
MKS$	7C5C	31836	FF-AF	255-175
MKD$	7C61	31841	FF-B0	255-176

ZONE DE COMMUNICATION



Structure

Rôle

Cette zone de 3200 octets, située en haut de la mémoire RAM 32K, est normalement interdite d'accès à l'utilisateur qui programme en Basic. Elle sert, comme son nom l'indique, de **zone d'échange** entre l'interpréteur Basic en ROM et le programme Basic en RAM.

Elle est, en fait, constituée de deux parties distinctes :

- la **zone des variables systèmes** où sont stockés essentiellement des pointeurs d'adresses (deux octets), des drapeaux (un octet), des tables et routines (quelques octets à quelques dizaines d'octets) et des variables internes (la connaissance précise de cette zone permet, à l'utili-

sateur, par des POKE appropriés, la reconfiguration des principaux paramètres des programmes Basic) ;

- la **table des vecteurs** (HOOK) dits "CROCHETS". Ces vecteurs sont, en fait, une interception en RAM de grandes routines Basic en ROM (non modifiables). Le déroutement par la zone RAM permet alors l'interception éventuelle de la routine et sa modification. Chaque vecteur contient cinq octets qui, sans interception, ont la valeur C9H (RET), c'est-à-dire un simple retour à la routine ROM. Lors d'une interception, un "JUMP" à l'adresse XXYY est réalisé, soit en code machine la suite de trois octets : YY,XX,C3.

F380H-FD99H
62336D-64921D
Longueur = 2586 octets

* *Recopie de la zone ROM 7F3FH-7FDBH dans la zone de communication.*
(1) Dépend des modèles.
(2) Ou en format signé ADD signé = (ADD non signé -65536).
Exemple : ADD signé = 62386 - 65536 = -3150.

Adresse DEC (2)	*Adresse HEX*	*Longueur (octets)*	*Rôle*	*Contenu initial * (HEX)-(DEC)*
62336-62340	F380-F384	5	Routine de lecture des SLOTS de la page 0.	D3-A8-5E-18-03
62341-62347	F385-F38B	7	Routine d'écriture des SLOTS de la page 0.	D3-A8-73-7A-D3-A8-C9
62348-62361	F38C-F399	14	Routine de saut à l'intérieur d'un SLOT.	D3-A8-08-CD-98-F3-08 F1-D3-A8-08-CD-DD-E9
62362-62381	F39A-F3AD	20	Table des adresses définies par l'instruction DEFUSRn= : occupe 10 x 2 octets. L'adresse de l'USRO se trouve en F39AH sous la forme classique des adresses du Z80 (partie basse en F39AH, partie haute en F39BH). Avant toute déclaration, chacune des adresses contient un envoi vers le message d'erreur ILLEGAL FUNCTION CALL.	OMS OPS 10 x (5A-47) Adresse = 475AH
62382	F3AE	1	Longueur de ligne par défaut en mode SCREEN 0. Vaut 37D par défaut.	25H = 37D (1)
62383	F3AF	1	Longueur de ligne par défaut en mode SCREEN 1. Vaut 29D par défaut.	1DH = 29D (1)
62384	F3B0	1	Longueur de ligne courante. Cette adresse est modifiée par l'instruction WIDTH (37D par défaut).	25H = 37D (1)
62385	F3B1	1	Longueur de page. C'est le nombre de lignes sur un écran. Vaut 24 par défaut.	18H = 24D
62386	F3B2	1	Vous pouvez modifier cette valeur par POKE et changer ainsi la taille de votre écran. Nombre de caractères pour TAB. Vaut 14 en standard.	0EH = 14D

LISTE DES VARIABLES SYSTEME PRINCIPALES DE LA ZONE DE COMMUNICATION

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle	Contenu initial* (HEX)		
			En mode SCREEN 0			
62387-62388	F3B3-F3B4	2	Adresse de la TNC	00H-00H (0000H)	→ BASE (0)	MODE TEXTE 24 x 40
62389-62390	F3B5-F3B6	2	Adresse de la TC	00H-00H (0000H)	→ BASE (1)	
62391-62392	F3B7-F3B8	2	Adresse de la TGC	00H-08H (0800H)	→ BASE (2)	
62393-62394	F3B9-F3BA	2	Adresse de la TAS	00H-00H (0000H)	→ BASE (3)	
62395-62396	F3BB-F3BC	2	Adresse de la TGS	00H-00H (0000H)	→ BASE (4)	
			En mode SCREEN 1			
62397-62398	F3BD-F3BE	2	Adresse de la TNC	00H-18H (1800H)	→ BASE (5)	MODE GRAPHIQUE I 24 x 32
62399-62400	F3BF-F3C0	2	Adresse de la TC	00H-20H (2000H)	→ BASE (6)	
62401-62402	F3C1-F3C2	2	Adresse de la TGC	00H-00H (0000H)	→ BASE (7)	
62403-62404	F3C3-F3C4	2	Adresse de la TAS	00H-1BH (1B00H)	→ BASE (8)	
62405-62406	F3C5-F3C6	2	Adresse de la TGS	00H-38H (3800H)	→ BASE (9)	
			En mode SCREEN 2			
62407-62408	F3C7-F3C8	2	Adresse de la TNC	00H-18H (1800H)	→ BASE (10)	MODE GRAPHIQUE II 192 x 256
62409-62410	F3C9-F3CA	2	Adresse de la TC	00H-20H (2000H)	→ BASE (11)	
62411-62412	F3CB-F3CC	2	Adresse de la TGC	00H-00H (0000H)	→ BASE (12)	
62413-62414	F3CD-F3CE	2	Adresse de la TAS	00H-1BH (1B00H)	→ BASE (13)	
62415-62416	F3CF-F3D0	2	Adresse de la TGS	00H-38H (3800H)	→ BASE (14)	

LISTE DES VARIABLES SYSTEME PRINCIPALES DE LA ZONE DE COMMUNICATION

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle	Contenu initial* (HEX)		
			En mode SCREEN 3			
62417-62418	F3D1-F3D2	2	Adresse de la TNC	00H-08H (0800H)	→ BASE (15)	MODE MULTICOLORE 48 x 64
62419-62420	F3D3-F3D4	2	Adresse de la TC	00H-00H (0000H)	→ BASE (16)	
62421-62422	F3D5-F3D6	2	Adresse de la TGC	00H-00H (0000H)	→ BASE (17)	
62423-62424	F3D7-F3D8	2	Adresse de la TAS	00H-1BH (1B00H)	→ BASE (18)	
62425-62426	F3D9-F3DA	2	Adresse de la TGS	00H-38H (3800H)	→ BASE (19)	

** Sous format OMS,OPS.*
( ) = Adresse dans VIDEORAM (0-3FFFH).

TAS = TABLE DES ATTRIBUTS DES SPRITES (lutins).
TC = TABLE DES COULEURS.
TGC = TABLE DU GENERATEUR DE CONFIGURATION (ou de CARACTERE).
TGS = TABLE DU GENERATEUR DES SPRITES.
TNC = TABLE DES NOMS DE CONFIGURATION.

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle	Contenu initial (1) (HEX) + (DEC)	
62427	F3D8	1	0 = pas de son sur les touches 1 = son *	01H = 1D	
62428	F3DC	1	Position courante verticale du curseur.	01H = 1D→Y	
62429	F3DD	1	Position courante horizontale du curseur.	01H = 1D→X	
62430	F3DE	1	1 = pas de fonction visible en bas d'écran (F1-F10). 0 = fonctions visibles.	00H = 0D→fonctions visibles	
62431-62438	F3DF-F3E6	8	Contenu des huit registres du VDP dans l'ordre 0 à 7.	00-E0-00-00-00-00-00-00	
62439	F3E7	1	Vaut 0 (drapeau).	00H = 0D	
62440	F3E8	1	Vaut 255 (0FFH) (drapeau).	FFH = 255D	
62441	F3E9	1	Couleur du texte (octet utilisé par COLOR).	0FH = 15D (blanc)	
62442	F3EA	1	Couleur du fond.	04H = 4D (bleu foncé)	
62443	F3EB	1	Couleur du bord.	04H = 4D (bleu foncé)	(2)
62444-62446	F3EC-F3EE	3	Contient C3 00 00 (JP 0000H).	C3-00-00	
62447-62449	F3EF-F3F1	3	Contient C3 00 00 (JP 0000H).	C3-00-00	
62450	F3F2	1	Octet attribut.	0FH = 15D	
62451-62452	F3F3-F3F4	2	Adresse de la table des QUEUES (pointeur).	59-F9→(F959H)	

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle	Contenu initial (1) (HEX) + (DEC)	
62453	F3F5	1	Contient FFH.	FFH = 255D	
62454	F3F6	1	Synchronisation du balayage des touches.	01H = 1D	
62455	F3F7	1	Contient 50D = 32H.	32H = 50D	
62456-62457	F3F8-F3F9	2	Adresse de l'octet courant à écrire dans le tampon clavier.	F0-FB→(FBF0H)	(3)
62458-62459	F3FA-F3FB	2	Adresse de l'octet courant à lire dans le tampon clavier.	F0-FB→(FBF0H)	(3)
62460-62478	F3FC-F40E	19	Ces 19 octets constituent les paramètres utiles aux fonctions de gestion de la cassette.	53-5C-26-2D-0F-25-2D-0E-16-1F-53-5C-26-2D-0F-00-01-00-01	
62479-62483	F40F-F413	5	Pointeur pour l'instruction RESUME NEXT.	3A-00-00-00-00	
62484	F414	1	Contient le numéro de la dernière erreur.	00H = 0D	

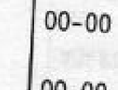
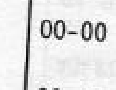
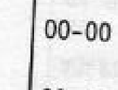
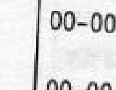
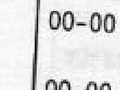
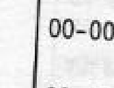
* *BEEP bref.*
(1) *Recopie de la zone ROM 7F3FH-7FDBH dans la zone de communication.*
(2) *Dépend des modèles (dans certains cas, 07 = cyan).*
(3) *Adresse de début du tampon clavier (buffer).*
( ) *→ adresse.*

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle	Contenu initial (HEX) + (DEC)
62485	F415	1	Contient la position de la tête de l'imprimante.	00H = 0D
62486	F416	1	Drapeau imprimante : 1 = imprimante, 0 = écran.	00H = 0D
62487	F417	1	0 = imprimante MSX, 1 = imprimante non MSX.	00H = 0D
62488	F418	1	Si ≠ 0, alors le caractère à sortir n'est pas codé.	00H = 0D
62489-62491	F419-F41B	2 + 1	Utilisé par la fonction VAL (adresse + caractère).	00-00-00
62492-62493	F41C-F41D	2	Numéro courant de la ligne en cours d'exécution.	FF-FF
62495-62812	F41F-F55C	318	Tampon pour le codage d'une ligne Basic (KRUNCH) *	-
62814-63071	F55E-F65F	258	Zone tampon pour le clavier.	-
63074	F662	1	Drapeau de l'instruction DIM (pas DIM=0).	00H = 0D
63075	F663	1	Drapeau qui indique le type de variable présente dans RA1 (DAC) (1).	02H = 2D
63076	F664	1	Type d'opérateur.	00H = 0D
63078-63079	F666-F667	2	Adresse du caractère courant dans le texte.	2A-F4→(F42AH)
63080	F668	1	Sauvegarde temporaire du code de l'instruction.	C4H

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle	Contenu initial (HEX) - (DEC)
63090-63091	F672-F673	2	Valeur supérieure de la mémoire utilisable par Basic. Cette valeur est modifiée par l'instruction CLEAR (2).	68-F1→(F168H)
63092-63093	F674-F675	2	Adresse supérieure du SP (pointeur de pile).	A0-F0→(F0A0H)
63094-63095	F676-F677	2	Adresse du début du texte du programme Basic (2).	01-80→(8001H)
63128-63129	F698-F699	2	Adresse du prochain octet disponible dans la table des chaînes (LITTERAL STRING POOL TABLE).	40-D7→(D740H)
63131-63132	F69B-F69C	2	Adresse du sommet de la LSPT.	68-F1→(F168H)
63137-63138	F6A1-F6A2	2	Pointeur pour l'instruction FOR.	00-00
63139-63140	F6A3-F6A4	2	Adresse de la dernière ligne DATA lue.	00-00
63141	F6A5	1	Drapeau pour FOR et USR.	00H = 0D
63142	F6A6	1	Drapeau pour INPUT et READ (0 si INPUT).	00H = 0D
63145	F6A9	1	Drapeau : mode programme ou mode direct (0 si mode direct).	00H = 0D

* *Point d'entrée en **ROM** du **CRUNCHER** : 42B2H.*
(1) *Voir **ZONE DE TRAVAIL DES REGISTRES**.*
(2) *Voir **ZONES MEMOIRES UTILISEES POUR LA PROGRAMMATION BASIC**.*

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle	Contenu initial (HEX) - (DEC)
63146	F6AA	1	Drapeau : 0 = AUTO, 1 = PAS AUTO.	00H = 0D
63147-63148	F6AB-F6AC	2	Numéro de la ligne courante (utilisé par AUTO).	00-00
63149-63150	F6AD-F6AE	2	Valeur de l'incrément entre deux lignes (AUTO).	00-00
63151-63152	F6AF-F6B0	2	Pointeur pour instruction RESUME (adresse).	1E-F4 → (F41EH)
63153-63154	F6B1-F6B2	2	Sauvegarde l'adresse de la PILE pour manipuler une erreur.	9E-F0 → (F09EH)
63155-63156	F6B3-F6B4	2	Pointeur du numéro de la ligne en erreur.	FF-FF
63157-63158	F6B5-F6B6	2	Pointeur du numéro de la ligne courante.	00-00
63159-63160	F6B7-F6B8	2	Pointeur pour l'instruction RESUME.	1E-F4 → (F41EH)
63161-63162	F6B9-F6BA	2	Numéro de la ligne du traitement d'erreur.	00-00
63163	F6BB	1	Drapeau : vaut 1 durant l'erreur et 0 après le traitement (RESUME).	00H = 0D
63166-63167	F6BE-F6BF	2	Numéro de ligne après STOP ou END (ancien numéro).	00-00
63168-63169	F6C0-F6C1	2	Adresse du dernier octet exécuté.	00-00
63170-63171	F6C2-F6C3	2	Adresse de la table des variables simples (1).	03-80 → (8003H)
63172-63173	F6C4-F6C5	2	Adresse de la table des variables tableaux (1).	03-80 → (8003H)

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle	Contenu initial (HEX) - (DEC)
63174-63175	F6C6-F6C7	2	Adresse du début de l'espace disponible (1).	03-80 → (8003H)
63176-63177	F6C8-F6C9	2	Pointe sur l'octet qui suit le dernier caractère en cours d'exécution.	00-80 → (8000H)
63178-63203	F6CA-F6E3	26	Table de déclaration des variables. Composée de 26 octets (1 par lettre de l'alphabet). Chaque octet contient un code qui détermine le type par défaut de chaque variable commençant par cette lettre. En standard, toutes les variables sont définies en double précision(8). (2)	26 octets 08H pour marque par défaut : double précision

(1) Voir *ZONES MEMOIRES UTILISEES POUR LA PROGRAMMATION BASIC.*
(2) Voir *TABLE DES INDICATEURS DE TYPE DE DONNEES ou TABLE DE DECLARATION DES VARIABLES.*

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle	Contenu initial (HEX)
63420-63427	F7BC-F7C3	8	Zone de stockage temporaire pour SWAP.	8 octets 00H
63428	F7C4	1	Drapeau : 0 = TROFF, 1 = TRON.	
63429 ↓	F7C5 ↓	↓	Début de la zone de travail des routines mathématiques.	-
63478-63485	F7F6-F7FD	8	Accumulateur primaire (encore appelé DAC) (1).	00-00-F8-F7-00-00-00-00
63559-63566	F847-F84E	8	Accumulateur secondaire (encore appelé ARG) (1).	00-00-00-00-00-00-00-00
63583 ↓	F85F ↓	↓	Début de la zone des paramètres pour la manipulation des fichiers.	-
63615-63774	F87F-F91E	160*	Contenu des touches fonctions (F1-F10).	Zone ROM 13A9H-1448H (2)
63776-63785	F920-F929**	10	Valeur courante des tables du VDP.	Voir **
63786-63787	F92A-F92B	2	Zone de travail pour le progiciel graphique.	00-00
63793-63816	F931-F948	23	Zone de travail pour l'instruction CIRCLE.	-
63817-63829	F949-F955	12	Zone de travail pour l'instruction PAINT.	-
63830-64395	F949-FB8B	565	Zone de travail pour l'instruction PLAY.	-
64432	FBB0	1	Démarrage à chaud si ≠ 0 (drapeau).	00H
64433	FBB1	1	≠ 0 si le Basic est en ROM (drapeau).	01H
64460	FBCC	1	Code du curseur.	- (3)
64462-64471	FBCE-FBD7	10	Etiquettes pour ON KEY GOSUB.	10 x 00H
64472	FBD8	1	Etiquette pour ON ... GOSUB.	00H

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle	Contenu initial (HEX)
64474-64484	FBDA-FBE4	11	Statut de l'ancienne touche.	8xFFH-7F-FF-FF
64485-64495	FBE5-FBEF	11	Statut de la nouvelle touche.	8xFFH-7F-FF-FF
64496-64535	FBF0-FC17	40	Tampon pour le codage de touche.	-
64584-64585	FC48-FC49	2	Adresse du début de la mémoire RAM.	00-80 → (8000H)
64586-64587	FC4A-FC4B	2	Adresse de fin de la mémoire RAM.	80-F3 → (F380H)
64666	FC9A	1	Contrôle d'interruption.	
64667	FC9B	1	Drapeau d'interruption.	00H

* *Dix groupes de 16 octets.*

*(1) Voir **ZONES DE TRAVAIL DES REGISTRES** (ZTR1-ZTR2) ou **REGISTRES AUXILIAIRES** (RA1-RA2) → RA1 = DAC, RA2 = AR6.*

*(2) Voir **TABLE DES VALEURS PAR DEFAUT DES TOUCHES DE FONCTION** (ROM).*

(3) Dépend du code du caractère sur lequel se trouve le curseur.

*** F920-F921 → adresse générateur de caractère en ROM (1BBF)*

F922-F923 → 00-00 (0000H) : base de TNC
F924-F925 → 00-08 (0800H) : base de TC
F926-F927 → 00-38 (3800H) : base de TGS
F928-F929 → 00-1B (1B00H) : base de TAS
} VIDEORAM

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle	Contenu initial (HEX) - (DEC)
64668	FC9C	1	Valeur Y de la manette de jeu.	00H = 0D
64669	FC9D	1	Valeur X de la manette de jeu.	00H = 0D
64672-64673	FCA0-FCA1	2	Valeur de l'intervalle pour ON INTERVAL GOSUB.	00-00
64674-64675	FCA2-FCA3	2	Compteur de INTERVAL.	45-D0 → (D045)
64678	FCA6	1	En tête de caractère graphique.	00H = 0D
64679	FCA7	1	Compteur de la séquence ESCAPE.	00H = 0D
64680	FCA8	1	Drapeau mode insertion.	00H = 0D
64681	FCA9	1	Drapeau curseur ON ou OFF.	00H = 0D
64682	FCAA	1	Caractère du curseur.	00H = 0D
64683	FCAB	1	Drapeau CAPS LOCK (FFH = 255D si CAPS LOCK, 00H si non).	00H = 0D
64686	FCAE	1	Drapeau chargement de programme Basic.	00H = 0D
64687	FCAF	1	Mode courant de l'écran.	00H = 0D
64688	FCB0	1	Ancien mode de l'écran.	00H = 0D
64690	FCB2	1	Couleur du contour pour PAINT.	00H = 0D
64691-64692	FCB3-FCB4	2	Position horizontale du curseur en graphique.	00-00
64693-64694	FCB5-FCB6	2	Position verticale du curseur en graphique.	00-00
64695-64696	FCB7-FCB8	2	Accumulateur graphique X.	00-00
64697-64698	FCB9-FCBA	2	Accumulateur graphique Y.	00-00

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle	Contenu initial (HEX) - (DEC)
64699	FCBB	1	Etiquette pour DRAW.	00H = 0D
64700	FCBC	1	Echelle pour DRAW.	00H = 0D
64701	FCBD	1	Angle pour DRAW.	00H = 0D
64702	FCBE	1	Drapeau entrée/sortie binaire.	00H = 0D
64703-64704	FCBF-FCC0	2	Pointeur de l'adresse d'exécution (OMS,OPS) d'un programme en code binaire (BSAVE).	00-00 —
64705-64708	FCC1-FCC4	4	Table des drapeaux pour extension de SLOTS (FFH = 255D si extension).	00-00-00-00
64709-64712	FCC5-FCC8	4	Etat courant pour chaque SLOT étendu.	0F-00-00
64713-64776	FCC9-FD08	64	Attributs pour chaque registre de SLOT étendu.	64 x 00H
64777-64904	FD09-FD88	128	Zone de travail spécifique à chaque SLOT.	128 x 00H
64905-64920	FD89-FD98	16	Nom de l'instruction étendue terminée par 0.	16 x 00H
64921	FD99	1	Identification ID pour une cartouche (0-3).	00H = 0D

FD9AH-FFC9H
64922D-65481D
Longueur = 560 octets*

Contenu initial

Chaque vecteur non intercepté contient cinq octets dont la valeur est C9 (RET) provoquant un retour à la routine appelante.

Déroutement

En cas d'interception, chaque vecteur contient alors C3 (JUMP) suivi de l'adresse de déroutement en format OMS,OPS.

Adresse DEC	*Adresse HEX*	*Longueur (octets)*	*Rôle*
64922-64926	FD9A-FD9E	5	Appel en OC4BH. VDP traitement des interruptions.
64927-64931	FD9F-FDA3	5	Appel en OC53H. VDP traitement des interruptions. Ce vecteur est appelé après la lecture du registre d'état du VDP.
64932-64936	FDA4-FDA8	5	Appel lors de l'écriture sur écran du caractère contenu dans A en mode TEXTE.
64937-64941	FDA9-FDAD	5	Appel lors de la mise à jour du curseur.
64942-64946	FDAE-FDB2	5	Appel lors de l'effacement du curseur.
64947-64951	FDB3-FDB7	5	Appel lors de l'affichage des fonctions F1-F10.
64952-64956	FDB8-FDBC	5	Appel lors de l'effacement des fonction F1-F10.
64957-64961	FDBD-FDC1	5	Appel lors du retour au mode TEXTE (32 ou 40) après un passage en mode graphique 2 ou multicolore.
64962-64966	FDC2-FDC6	5	Appel lors de la lecture d'un caractère.
64967-64971	FDC7-FDCB	5	Appel en 071E. Appel lors de l'initialisation du VDP (chargement de la table des caractères...).

* *soit 112 vecteurs de cinq octets.*

Adresse DEC	*Adresse HEX*	*Longueur (octets)*	*Rôle*
64972-64976	FDCC-FDD0	5	Appel lors de la lecture clavier au moment où l'accumulateur contient dix fois le numéro de la ligne de la touche enfoncée + le numéro de la colonne de cette touche.
64977-64981	FDD1-FDD5	5	Appel en OF10H. Appel avant de convertir un caractère émis par le clavier d'après la table située en 1003H.
64982-64986	FDD6-FDDA	5	Appel lors du traitement d'une interruption non masquable.
64987-64991	FDDB-FDDF	5	Appel lors de l'impression de message système. Ce vecteur sert à l'insertion d'une carte 80 colonnes.
64992-64996	FDE0-FDE4	5	Appel lors de l'impression d'un ? suivi d'un INPUT. Ce vecteur est intercepté en mode 80 colonnes.
64997-65001	FDE5-FDE9	5	Appel lors de l'INPUT. Ce vecteur est intercepté en 80 colonnes.
65002-65006	FDEA-FDEE	5	Appel lors du traitement des instructions ON GOTO, ON GOSUB.
65007-65011	FDEF-FDF3	5	Appel lors de l'instruction DSKO$. Intercepté par SED.
65012-65016	FDF4-FDF8	5	Appel lors de l'instruction SET. Intercepté par SED.
65017-65021	FDF9-FDFD	5	Appel lors de l'instruction NAME. Intercepté par SED.
65022-65026	FDFE-FE02	5	Appel lors de l'instruction KILL. Intercepté par SED.
65027-65031	FE03-FE07	5	Appel lors de l'instruction IPL. Intercepté par SED.
65032-65036	FE08-FE0C	5	Appel lors de l'instruction COPY. Intercepté par SED.
65037-65041	FE0D-FE11	5	Appel lors de l'instruction CMD. Intercepté par SED.

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle
65042-65046	FE12-FE16	5	Appel lors de l'instruction DSKF. Intercepté par SED.
65047-65051	FE17-FE1B	5	Appel lors de l'instruction DSKI$. Intercepté par SED.
65052-65056	FE1C-FE20	5	Appel lors de l'instruction ATTR$. Intercepté par SED.
65057-65061	FE21-FE25	5	Appel lors de l'instruction LSET. Intercepté par SED.
65062-65066	FE26-FE2A	5	Appel lors de l'instruction RSET. Intercepté par SED.
65067-65071	FE2B-FE2F	5	Appel lors de l'instruction FIELD. Intercepté par SED.
65072-65076	FE30-FE34	5	Appel lors de la fonction MKI$. Intercepté par SED.
65077-65081	FE35-FE39	5	Appel lors de la fonction MKS$. Intercepté par SED.
65082-65086	FE3A-FE3E	5	Appel lors de la fonction MKD$. Intercepté par SED.
65087-65091	FE3F-FE43	5	Appel lors de la fonction CVI. Intercepté par SED.
65092-65096	FE44-FE48	5	Appel lors de la fonction CVS. Intercepté par SED.
65097-65101	FE49-FE4D	5	Appel lors de la fonction CVD. Intercepté par SED.
65102-65106	FE4E-FE52	5	Vecteur intercepté par le SED. Utilisé lors du positionnement sur un fichier.
65107-65111	FE53-FE57	5	Appel lors du positionnement d'un pointeur sur un fichier ouvert.
65112-65116	FE58-FE5C	5	Appel lors de l'instruction OPEN. Intercepté par SED.
65117-65121	FE5D-FE61	5	Appel lors de KILL, LOAD, MERGE. Intercepté par SED.
65122-65126	FE62-FE66	5	Appel lors de l'instruction CLOSE. Intercepté par SED.

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle
65127-65131	FE67-FE6B	5	Appel lors de l'instruction MERGE. Intercepté par SED.
65132-65136	FE6C-FE70	5	Appel au début d'une instruction SAVE (SED).
65137-65141	FE71-FE75	5	Appel dans le corps d'une instruction SAVE (SED).
65142-65146	FE76-FE7A	5	Appel à la fin d'une instruction SAVE (SED).
65147-65151	FE7B-FE7F	5	Appel lors de l'instruction FILES. Intercepté par SED.
65152-65156	FE80-FE84	5	Appel lors de l'instruction GET ou PUT (SED).
65157-65161	FE85-FE89	5	Appel lors de sortie sur un fichier (SED).
65162-65166	FE8A-FE8E	5	Appel lors du test du DEVICE. Permet d'installer d'autres DEVICES.
65167-65171	FE8F-FE93	5	Appel lors de l'instruction INPUT$.
65172-65176	FE94-FE98	5	Appel lors de la rencontre d'une fonction SED (LOC, LOF, EOF, FPOS).
65177-65181	FE99-FE9D	5	Appel lors de la fonction LOC. Intercepté par le SED.
65182-65186	FE9E-FEA2	5	Appel lors de la fonction LOF. Intercepté par le SED.
65187-65191	FEA3-FEA7	5	Appel lors de la fonction EOF. Intercepté par le SED.
65192-65196	FEA8-FEAC	5	Appel lors de la fonction FPOS. Intercepté par le SED.
65197-65201	FEAD-FEB1	5	Vecteur utilisé pour interfacer le SED.
65202-65206	FEB2-FEB6	5	Appel au début de l'analyse du nom du DEVICE.
65207-65211	FEB7-FEBB	5	Appel si le nom n'est pas dans la table des DEVICES.
65212-65216	FEBC-FEC0	5	Appel si le nom est effectivement celui d'un DEVICE.

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle
65217-65221	FEC1-FEC5	5	CE VECTEUR N'EST PAS UTILISE.
65222-65226	FEC6-FECA	5	Appel lors du traitement d'un DEVICE non DISQUE.
65227-65231	FECB-FECF	5	Appel lors du NEW ou du RUN.
65232-65236	FED0-FED4	5	Appel lors de l'initialisation de la table des variables.
65237-65241	FED5-FED9	5	Appel lors de l'initialisation de la table des variables (boucle).
65242-65246	FEDA-FEDE	5	Appel lors du nettoyage des trames FOR et GOSUB.
65247-65251	FEDF-FEE3	5	Appel lors du test de l'existence d'un fichier.
65252-65256	FEE4-FEE8	5	Appel lors de la sortie d'un caractère sur écran ou imprimante.
65257-65261	FEE9-FEED	5	Appel lors de l'impression d'un CR suivi d'un LF.
65262-65266	FEEE-FEF2	5	Appel lors d'INPUT d'un DEVICE.
65267-65271	FEF3-FEF7	5	Appel lors des fonctions graphiques (LINE, CIRCLE,...).
65272-65276	FEF8-FEFB	5	Appel à la fin de l'exécution d'un programme.
65277-65281	FEFD-FF01	5	Appel lors de l'impression d'un message d'erreur.
65282-65286	FF02-FF06	5	Appel à la fin de l'impression du message d'erreur.
65287-65291	FF07-FF0B	5	Appel lors de l'impression du message OK et du retour au mode d'entrée.
65292-65296	FF0C-FF10	5	Appel à l'entrée de l'interpréteur.
65297-65301	FF11-FF15	5	Appel lors de l'exécution en mode direct.
65302-65306	FF16-FF1A	5	Appel à la fin de l'interprétation d'une instruction.
65307-65311	FF1B-FF1F	5	Appel à la fin de l'interprétation.

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle
65312-65316	FF20-FF24	5	Appel à l'entrée du CRUNCHER (routine de transformation d'une ligne Basic en code de représentation des instructions).
65317-65321	FF25-FF29	5	Appel lors du début de la recherche d'une instruction dans la table alphabétique.
65322-65326	FF2A-FF2C	5	Appel lors de la découverte d'un mot réservé dans la phase de CRUNCH.
65327-65331	FF2F-FF33	5	Appel lorsque le mot réservé est suivi d'un numéro de ligne (GOTO THEN...).
65332-65336	FF34-FF38	5	Le mot n'est pas réservé.
65337-65341	FF39-FF3D	5	Ce vecteur permet l'installation d'une autre routine mathématique.
65342-65346	FF3E-FF42	5	Appel au début d'une nouvelle instruction.
65347-65351	FF43-FF47	5	Appel lors des instructions de déroutement (GOTO, IF...).
65352-65356	FF48-FF4C	5	Appel lors de la saisie d'un caractère.
65357-65361	FF4D-FF51	5	Appel lors du traitement de l'instruction RETURN.
65362-65366	FF52-FF56	5	Appel lors de l'instruction PRINT.
65367-65371	FF57-FF5B	5	Appel dans le corps du traitement de l'instruction PRINT.
65372-65376	FF5C-FF60	5	Appel à la fin du traitement d'une instruction PRINT.
65377-65381	FF61-FF65	5	Appel lors du traitement d'un DATA ou d'un INPUT incorrect.
65382-65386	FF66-FF6A	5	Appel lors de l'évaluation d'une formule.
65387-65391	FF6B-FF6F	5	Permet l'installation d'une autre routine mathématique lors de l'évaluation de formule.

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle
65392-65396	FF70-FF74	5	Appel lors de l'évaluation d'une expression.
65397-65401	FF75-FF79	5	Appel lors de l'évaluation des fonctions transcendantes. Ce vecteur permet l'installation d'une autre routine mathématique.
65402-65406	FF7A-FF7E	5	Appel à la fin de l'évaluation des fonctions transcendantes.
65407-65411	FF7F-FF83	5	Appel lors du traitement de l'instruction MID$.
65412-65416	FF84-FF85	5	Appel lors de l'instruction WIDTH.
65417-65421	FF89-FF8D	5	Appel lors de l'instruction (L)LIST.
65422-65426	FF8E-FF92	5	Appel lors de l'instruction LIST au moment de convertir le code en mot-clé.
65427-65431	FF93-FF97	5	Appel lors de l'instruction POKE. Ce vecteur permet l'installation d'une autre routine mathématique.
65432-64436	FF98-FF9C	5	Appel lors de la conversion d'un numéro de ligne en pointeur et inversement.
65437-65441	FF9D-FFA1	5	Appel avant la recherche d'une place libre pour une nouvelle chaîne de caractères.
65442-65446	FFA2-FFA6	5	Appel lors de la lecture d'un nom de variable à la position courante dans le texte.
65447-65451	FFA7-FFAB	5	Ce vecteur est utilisé par le BIOS en 145H et n'a pas d'utilité en configuration normale.
65452-65456	FFAC-FFB0	5	Ce vecteur est utilisé par le CALL en 148H et n'a pas d'utilité en configuration normale.
65457-65461	FFB1-FFB5	5	Appel lors du traitement d'erreur.

Adresse DEC	Adresse HEX	Longueur (octets)	Rôle
65462-65466	FFB6-FFBA	5	Appel lors de l'impression sur l'imprimante.
65467-65471	FFBB-FFBF	5	Appel lors du test du statut de l'imprimante.
65472-65476	FFC0-FFC4	5	Appel lors de l'instruction SCREEN.
65477-65481	FFC5-FFC9	5	Appel lors de l'instruction PLAY.

Rôle

Le "**Pointeur**" utilise la technique de **l'adressage indirect**, c'est-à-dire que le contenu d'une adresse n'est pas un octet de donnée, mais un octet représentatif d'une autre adresse sous le format OMS, OPS. Cette technique, dite de "pointage" est utilisée pour repérer une zone mémoire susceptible de changer à partir d'un point mémoire fixe.

* *En version 32K RAM (C001H en version 16K RAM) à l'initialisation.*

( ) *Contenu du pointeur : AD DEC = (OMS + OPS x 256)*

** *Accessible à l'utilisateur par CLEAR n,p n = 200 D = C8H. Par défaut, p = valeur mémoire utilisateur à protéger.*

F6CAH à F6E3H
63178D à 63203D

Adresse HEX	Adresse DEC	Lettre INDEX	Contenu initial HEX
F6CA	63178	A	08
F6CB	63179	B	08
F6CC	63180	C	08
F6CD	63181	D	08
F6CE	63182	E	08
F6CF	63183	F	08
F6D0	63184	G	08
F6D1	63185	H	08
F6D2	63186	I	08
F6D3	63187	J	08
F6D4	63188	K	08
F6D5	63189	L	08
F6D6	63190	M	08
F6D7	63191	N	08
F6D8	63192	O	08
F6D9	63193	P	08
F6DA	63194	Q	08
F6DB	63195	R	08
F6DC	63196	S	08
F6DD	63197	T	08
F6DE	63198	U	08
F6DF	63199	V	08
F6E0	63200	W	08
F6E1	63201	X	08
F6E2	63202	Y	08
F6E3	63203	Z	08

Rôle

Cette table de 26 octets mémorise l'indicateur de type de donnée "ITD" pour chacune des variables commençant par une lettre donnée ayant subi une déclaration Basic de type DEF :

DEFINT → ITD = 02H (E)
DEFSTR → ITD = 03H (CC)
DEFSNG → ITD = 04H (SP)
DEFDBL → ITD = 08H (DP)

Les valeurs initiales, contenues dans la table, sont 08H car les valeurs implicites, sans déclaration de type, sont en "double précision".

ZTR1 : F7F6H à F7FDH	
RA1 : 63478D à 63485D	
ZTR2 : F847H à F84EH	
RA2 : 63559D à 63566D	

Rôle

Ces registres viennent servir de support aux registres internes du Z80 (AF, BC, HL, DE) qui ne permettent que des additions ou des soustractions entières sur 8 et 16 bits. MSX utilise, en format DP, un ensemble de 8 octets (4 octets en SP). Ils peuvent être considérés comme des accumulateurs externes utilisés par les registres HL, DE, chacun d'entre eux ayant une taille de 8 octets autorisant le stockage des 8 octets du format DP.

Structure de ZTR1-RA1 (DAC)

E = Entier
SP = Simple précision
DP = Double précision

Adresse HEX	*Adresse DEC*	*Contenu E*	*Contenu SP*	*Contenu DP*
F7F6	63478		EXPOSANT *	EXPOSANT *
F7F7	63479		MANTISSE 1er octet	MANTISSE 1er octet
F7F8	63480		2e octet	2e octet
F7F9	63481		3e octet	3e octet
F7FA	63482			4e octet
F7FB	63483			5e octet
F7FC	63484			6e octet
F7FD	63485			7e octet

* *Contient également le signe de la mantisse (se reporter au format de stockage des nombres signés en mémoire).*

Structure de ZTR2-RA2 (ARG)

Adresse HEX	*Adresse DEC*	*Contenu E*	*Contenu SP*	*Contenu DP*
F847	63559		EXPOSANT *	EXPOSANT *
F848	63560		MANTISSE 1er octet	MANTISSE 1er octet
F849	63561		2e octet	2e octet
F84A	63562		3e octet	3e octet
F84B	63563			4e octet
F84C	63564			5e octet
F84D	63565			6e octet
F84E	63566			7e octet

* *Contient également le signe de la mantisse (se reporter au format de stockage des nombres signés en mémoire).*

Utilisation conjuguée des registres internes et externes pour les opérations arithmétiques (2 variables)

ITD = Indicateur de Type de Donnée

*Opération**	*Registre Source*	*Registre Destination (résultat)*	*Nombre octets résultat*	*ITD (HEX)*
Addition E	DE + HL	HL	2	02
Soustraction E	DE - HL	HL	2	02
Multiplication E	DE * HL	HL	2	02
Division E	DE / HL	HL *	2	02
Addition SP	RA1 + RA2**	RA1	4	04
Soustraction SP	RA1 - RA2	RA1	4	04
Multiplication SP	RA1 * RA2	RA1	4	04
Division SP	RA1 / RA2	RA1	4	04
Addition DP	RA1 + RA2**	RA1	8	08
Soustraction DP	RA1 - RA2	RA1	8	08
Multiplication	RA1 * RA2	RA1	8	08
Division	RA1 / RA2	RA1	8	08

* reste dans DE
** premier nombre n1 va dans RA2 si n1<n2 ou n1 même décade que n2 (n1>n2) sinon n2 dans RA2

Utilisation des registres externes (RA1) pour les opérations mathématiques (1 variable)

- Quelle que soit la routine utilisée (SQR, LOG, EXP)

Registre Source = RA1 (E,SP,DP)
Registre Destination = RA1 (E,SP,DP)

+ RA2 contient également la valeur en DP du résultat.

- Du fait que les routines mathématiques utilisent des opérations arithmétiques nécessitent la connaissance de l'ITD de l'argument de RA1, celui-ci se trouve en F663H ou 63075D.

OMS ** OPS

(F676H)-(F677H) (63094D)(63095D)

INTERPRETEUR BASIC

TABLE DES INSTRUCTIONS DE PROGRAMME (TIP) — Evolution automatique dynamique de la zone du bas vers le haut de la mémoire

(F6C2H)-(F6C3H) (63170D)(63171D)

TABLE DES VARIABLES SIMPLES (TVS) — Evolution automatique dynamique de la zone du haut vers le bas de la mémoire

(F6C4H)-(F6C5H) (63172D)(63173D)

TABLE DES VARIABLES DIMENSIONNEES (ou TABLEAUX) (TVT) — Evolution avec contrôle utilisateur

(F6C6H)-(F6C7H) (63174D)(63175D)

ZONE MEMOIRE LIBRE (ZML)

(F674H)-(F675H) (63092D)(63093D)

PILE OPERATIONNELLE (PO)

[(F672H)-(F673H)]-n
[(63090D)(63091D)]-n

ZONE * CHAINE DE CARACTERES (ZCC)

(F672H)-(F673H) (63090D)(63091D)

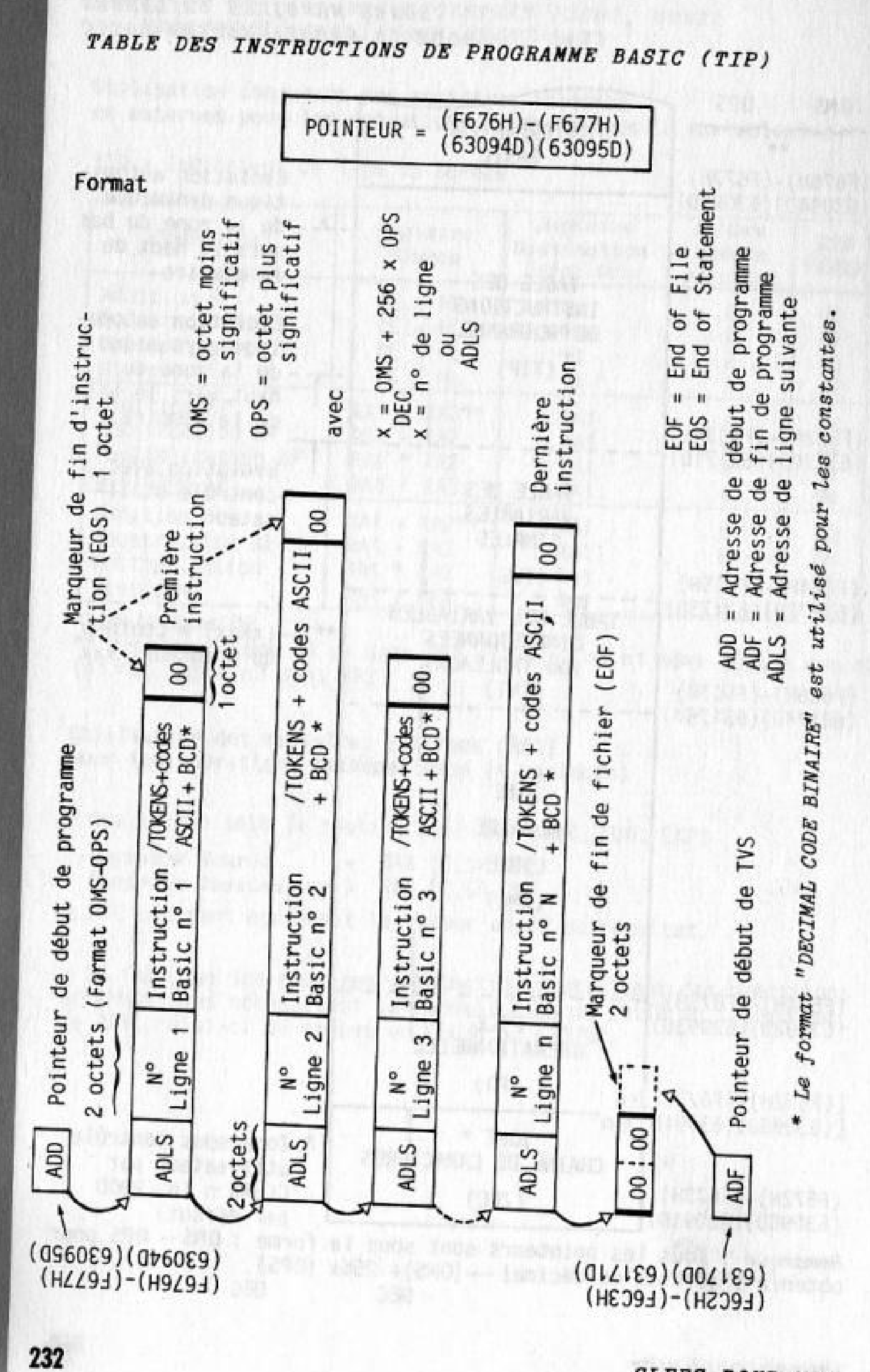
** → (xxxx) = contenu du pointeur xxxx

* Zone sous contrôle utilisateur par CLEAR n (n = 200D par défaut)

Remarque : Tous les pointeurs sont sous la forme : OMS - OPS pour obtenir l'adresse en décimal → (OMS)+ 256x (OPS).
DEC DEC

POINTEUR = (F676H)-(F677H) (63094D)(63095D)

Format



OMS = octet moins significatif

OPS = octet plus significatif

avec

x_{DEC} = OMS + 256 x OPS

x = n° de ligne ou ADLS

EOF = End of File
EOS = End of Statement

ADD = Adresse de début de programme
ADF = Adresse de fin de programme
ADLS = Adresse de ligne suivante

* *Le format "DECIMAL CODE BINAIRE" est utilisé pour les constantes.*

Exemple 1

(F676H)-(F677H) 01 80 Adresse de début de programme (sous RAM 32K)

N° de ligne — TOKENS — EOR = EOS

8001	1B	80	32	00	8F	45	58	45	4D	50	4C	45	20	44	45	20	50	52	4F	47	52	41	4D	4D	45	00
801B	35	80	64	00	91	22	44	45	42	55	54	20	44	45	20	50	52	4F	47	52	41	4D	4D	45	22	00
8035	41	80	C8	00	82	49	EF	11	D9	31	64	00														
8041	5C	80	FA	00	91	22	4D	49	4C	49	45	55	20	44	45	20	50	52	4F	47	52	41	4D	4D	45	22	00
805C	63	80	2C	01	83	49	00																			
8063	7B	80	90	01	91	22	46	49	4E	20	44	45	20	50	52	4F	47	52	41	4D	4D	45	22	00		
807B	00	00	Dernier octet de programme + 1																							

EOF — ADLS — (F6C2H)-(F6C3H) 7D80 Adresse de début de TVS (Table des Variables Simples)

```
50  REM    EXEMPLE DE PROGRAMME **
100 PRINT  "DEBUT DE PROGRAMME"
200 FOR I  =  0 TO 100
250 PRINT  "MILIEU DE PROGRAMME"
300 NEXT I
400 PRINT  "FIN DE PROGRAMME"
```

** Les espaces entre mots du programme (hors texte entre quotes) ne sont donnés que pour la lisibilité du programme, ils ne font pas partie intégrante de l'exemple.

Exemple 2

```
10 A=INT(2.44)
20 IF A<2 THEN 40
:
```

Pointeur début TIP (F676H) (F677H) → Début TIP

Adresse	Contenu TIP En Hex	Contenu TIP En Dec	
8001H	11	17	Adresse de l'instruction suivante (8011H)
8002H	80	128	
8003H	0A	10	N° de ligne = 10D = 0AH
8004H	00	00	
8005H	41	65	ASCII → A
8006H	EF	239	TOKEN pour opérateur "="
8007H	FF	255	TOKEN pour MOT-CLE "INT"
8008H	85	133	
8009H	28	40	ASCII → (
800AH	1D	29	Equivalent BCD de la constante 2.44
800BH	41	65	65 → signe de la mantisse + signe de l'exposant + valeur de l'exposant
800CH	24	36	24H,40H,00H → valeur de la mantisse
800DH	40	64	
800EH	00	00	
800FH	29	41	ASCII →)
8010H	00	00	→ Marqueur de fin d'instruction (EOS)
8011H	1E	30	Adresse de l'instruction suivante (801EH)
8012H	80	128	
8013H	14	20	N° de ligne = 20D = 14H
8014H	00	00	
8015H	8B	139	TOKEN pour MOT-CLE "IF"
8016H	41	65	ASCII → A
8017H	F0	240	TOKEN pour opérateur "<"
8018H	13	19	Equivalent de la constante 2
8019H	DA	218	TOKEN pour MOT-cle "THEN"
801AH	0E	14	Equivalent binaire signé de la constante 40 (0E = **repère de ligne Basic**) X DEC = OMS + 256 * OPS
801BH	28	40	
801CH	00	00	
801DH	00	00	→ Marqueur de fin d'instruction (EOS)
801EH, 801FH	SUITE TIP		

POINTEUR = F6C2H-F6C3H 63170D-63171D

Format

Type	*N° octet (DEC)*	*Contenu*	*Capacité*
NOMBRE ENTIER 5 OCTETS ×	1	Indicateur type Entier = 02H	-32768≤ N≤ + 32767
	2	Premier caractère du nom de la variable (codage ASCII)	
	3	Second caractère du nom de la variable (codage ASCII)	
	4	Octet le moins significatif (OMS) de la variable entière	
	5	Octet le plus significatif (OPS) de la variable entière	
NOMBRE SIMPLE PRECISION 7 OCTETS ×	1	Indicateur type "Simple Précision" = 04H	N≥ - 9,99999 10^{62}
	2	Premier caractère du nom de la variable (codage ASCII)	
	3	Second caractère du nom de la variable (codage ASCII)	
	4	Octet représentant (exposant + signe + signe de la mantisse) (EXP)	N≤ + 9,99999 10^{62}
	5	1er octet de la mantisse de la variable (BCD1)	
	6	2e octet de la mantisse de la variable(BCD2)	
	7	3e octet de la mantisse de la variable (BCD3)	

Représentation normalisée : $N = M.10^{E}$ - M = Mantisse - E = Exposant - Pour SP et DP.

× Se reporter à "**Format de stockage en mémoire des nombres signés**".

Type	N° octet (DEC)	Contenu	Capacité
NOMBRE DOUBLE PRECISION 11 OCTETS *	1	Indicateur type "Double Précision" = 08H	
	2	Premier caractère du nom de la variable (codage ASCII)	N > - 9,9999999999 999 10^{62}
	3	Second caractère du nom de la variable (codage ASCII)	
	4	Octet représentant (valeur exposant + signe exposant + signe de la mantisse) (EXP)	
	5	1er octet de la mantisse de la variable (BCD1)	N < + 9,9999999999 999 10^{62}
	6	2e octet de la mantisse de la variable (BCD2)	
	7	3e octet de la mantisse de la variable (BCD3)	
	8	4e octet de la mantisse de la variable (BCD4)	
	9	5e octet de la mantisse de la variable (BCD5)	
	10	6e octet de la mantisse de la variable (BCD6)	
	11	7e octet de la mantisse de la variable (BCD7)	
VARIABLE CHAINE DE CARACTERES 6 OCTETS *	1	Indicateur type "Chaîne de Caractères" = 03H	255 caractères
	2	Premier caractère du nom de la variable (codage ASCII)	
	3	Second caractère du nom de la variable (codage ASCII)	
	4	Longueur de la chaîne de caractères	
	5	OMS de l'adresse du premier caractère de la chaîne	
	6	OPS de l'adresse du premier caractère de la chaîne	

* Se reporter à "**Format de stockage en mémoire des nombres signés**".

Exemple

Variable	N° octet	Contenu HEX	Contenu DEC	Signification
A%=2583	1	02	2	Indicateur type Entier = 2
	2	41	65	Premier caractère du nom de la variable 41H→A
	3	00	0	Second caractère du nom de la variable 00H→non existant
	4	17	23	OMS de la valeur de la variable } Résultat =
(E)	5	0A	10	OPS de la valeur de la variable } 23+10x256 = 2583
X3!=2,5	1	04	4	Indicateur type SP = 4
	2	58	88	Premier caractère du nom de la variable 58H→X
	3	33	51	Second caractère du nom de la variable 33H→3
(SP)	4	41	65	$2{,}5 = 0{,}25 . 10^1 \rightarrow S_1=0,\ S_2=1$; Valeur de l'exposant = 1 → [0|1|0|0|0|0|0|1] = 41H = 65D
	5	25	37	Valeur de la mantisse : 1 octet [2|5], 1 octet [0|0], 1 octet [0|0] (3 octets)
	6	00	0	2,5 → code en BCD
	7	00	0	2 → [0|0|1|0|0|1|0|1] ← 5
ZZ = - 6,25	1	08	8	Indicateur type DP = 8
	2	5A	90	Premier caractère du nom de la variable 5AH→Z
	3	5A	90	Second caractère du nom de la variable 5AH→Z
	4	C1	193	$-6{,}25 = -0{,}625.10^1 \quad S_1=1,\ S_2=1$; Valeur de l'exposant = 1 → [1|1|0|0|0|0|0|1] = 193D=C1H
(DP)	5	62	98	Valeur de la mantisse : 1 octet [6|2] [5|0] [0|0] [0|0] [0|0] [0|0] [0|0] (7 octets)
	6	50	80	6,25 → codé en BCD
	7	00	0	6 → [0|1|1|0|0|0|1|0] ← 2 ; 5 → [0|1|0|1]
	8	00	0	
	9	00	0	
	10	00	0	
	11	00	0	

S1 = signe de la mantisse (0 si > 0, 1 si < 0) - **S2** = signe de l'exposant (0 si < 0, 1 si > 0).

Exemple

Variable	N° octet	Contenu HEX	Contenu DEC	Signification
S1$="ABCD"	1	03	3	Indicateur type CC = 3
	2	53	83	Premier caractère du nom de la variable 53H→S
	3	31	49	Second caractère du nom de la variable 31H→1
	4	04	4	Longueur de la chaîne de caractères = 4
	5	0A	10	OMS de l'adresse du premier caractère de la chaîne }
(CC)	6	80	128	OPS de l'adresse du premier caractère de la chaîne }
				Adresse = 800AH ou (10 + 128 x 256)D = 32778

* PRINT PEEK (32778) → 65 (valeur ASCII décimale du caractère "A" premier élément de la chaîne)

Procédure de décodage (→ décimal)

Type Variable	Etape	Opération (sur valeurs DEC)	Résultat
ENTIER (E)	1	- Sous Basic, faire VARPTR (nom de la variable)	- Adresse EXP (DEC)
	2	- Faire PEEK (adresse EXP) et PEEK (adresse EXP+1)	- OMS, OPS (DEC)
	3	- Multiplier la valeur de OPS par 256	- 256 x OPS
	4	- Calculer la valeur de N = OMS + 256 x OPS	- Résultat final (DEC)
SIMPLE PRECISION (SP)	1	- Sous Basic, faire VARPTR (nom de la variable)	- Adresse EXP (DEC)
	2	- Faire PEEK (adresse EXP)	- EXP
	3	- Calculer le **signe de la mantisse** en faisant LEFT$ (BIN$(PEEK(adresse EXP)),1)→bit 7	- **Signe mantisse** >0 si résultat = 0 <0 si résultat = 1
	4	- Calculer le **signe de l'exposant** en faisant MID$ (BIN$(PEEK(adresse EXP)),2,1)→(bit 6)	- **Signe exposant** >0 si résultat = 1 <0 si résultat = 0
	5	- Calculer la **valeur de l'exposant** en faisant MID$ (BIN$(PEEK(adresse EXP)),3,6)→bit 5 à bit 0, puis &B du résultat	- **Valeur de l'exposant** codée sur 6 bits en binaire (-10^{63} à 10^{63})
	6	- Faire PEEK (adresse EXP+1) à PEEK (adresse EXP+3)	- Adresses de la valeur de la mantisse (BCD1 à BCD3)
	7	- Calculer la valeur de la mantisse en faisant, pour chaque octet : MID$(BIN$(PEEK(adresse EXP+n)),1,4), puis MID$(BIN$(adresse EXP+n)),5,4) et finalement &B de ces deux résultats.	- **Valeur de la mantisse** sur trois octets (un chiffre par ½ octet) soit six chiffres significatifs

* *Le chiffre décimal le plus à gauche est stocké dans BCD1, le suivant dans BCD2, et ainsi de suite jusqu'à BCD3.*

Procédure de décodage

Type Variable	Etape	Opération (sur valeur DEC)	Résultat
DOUBLE PRECISION (DP)	1	- Sous Basic, faire VAPRTR (nom de la variable)	- Adresse EXP (DEC)
	2	- Faire PEEK (adresse EXP)	- EXP
	3	- Calculer le **signe de la mantisse** en faisant LEFT$(BIN$(PEEK(adresse EXP)),1)→bit 7	- **Signe mantisse** >0 si résultat = 0 <0 si résultat = 1
	4	- Calculer le **signe de l'exposant** en faisant MID$(BIN$(PEEK(adresse EXP)),2,1)→bit 6	- **Signe exposant** >0 si résultat = 1 <0 si résultat = 0
	5	- Calculer **la valeur de l'exposant** en faisant MID$(BIN$(PEEK(adresse EXP)),3,6), puis &B du résultat → bit 5 à bit 0	- **Valeur de l'exposant** codé sur six bits en binaire non signé (-1063 ou 1063)
	6	- Faire PEEK (adresse EXP+1) à PEEK (adresse EXP+7)	- Adresses de la valeur de la mantisse (BCD1 à BCD7)
	7	- Calculer la valeur de la mantisse en faisant, pour chaque octet constitutif, PEEK(ADR EXP+1), à PEEK(ADR EXP+7) : MID$(BIN$(PEEK(ADR EXP+n)),1,4)→1er ½ octet puis, MID$(BIN$(PEEK(ADR EXP+n)),5,4)→2e ½ octet et finalement &B de ces deux résultats.	- Valeur de la mantisse codée sur sept octets en BCD sous la forme d'un chiffre pour ½ octet, soit 14 chiffres significatifs.

* *Le chiffre décimal le plus à gauche se trouve dans BCD1, le suivant dans BCD2, et ainsi de suite jusqu'à BCD7.*

Procédure de décodage

Type Variable	Etape	Opération (sur valeurs DEC)	Résultat
CHAINE DE CARACTERES (CC)	1	- Sous Basic, faire VARPTR (nom de la variable)	- Adresse de la longueur de CC
	2	- Faire PEEK (adresse longueur CC +1) et PEEK (adresse longueur CC +2).	- OMS et OPS de l'adresse du 1e caractère de CC
	3	- Faire K = OMS + 256 x OPS.	- Adresse début CC
	4	- Faire PEEK (K) jusqu'à PEEK (K + longueur de chaîne de caractères -1).	- Valeurs ASCII des différents caractères = X
	5	- Faire PRINT CHR$(X)	- Affichage caractère

Remarque : Attention, le nom de la variable doit avoir été déclaré préalablement à l'utilisation de VARPTR, sinon une erreur "ILLEGAL FUNCTION CALL" sera retournée (appel de fonction invalide).

Décodage ($N = M \times 10^E$)

Opération	Résultat
DECODAGE EN DECIMAL DE LA VARIABLE STOCKEE SOUS LA FORME 41 25 00 00 (HEX) EXP BDC1 BCD2 BCD3 65 37 0 0 (DEC)	Exposant >0 Mantisse >0, Valeur exposant
- Ecriture de EXP en binaire - Le nombre peut être écrit : $M.10^1$ (M=MANTISSE) avec M< 1 (point décimal à gauche). - Deux premiers digits du nombre contenus dans BCD1 (25H).	→ . 0 1 0 0 0 0 0 1 → 65 = 1 − 10^1 E = 1 M BCD1 → 0 0 1 0 0 1 0 1 2 5
- Deux digits suivants du nombre contenus dans BCD2 (00H).	BCD2 → 0 0 0 0 0 0 0 0 0 0
- Deux derniers digits contenus dans BCD3.	BCD3 → 0 0 0 0 0 0 0 0 0 0 N! = $0,250000 \times 10^1$
DECODAGE EN DECIMAL DE LA VARIABLE STOCKEE SOUS LA FORME C0 37 50 00 (HEX) EXP BCD1 BCD2 BCD3 192 55 80 0 (DEC)	Exposant >0 Mantisse <0, Valeur exposant
- Ecriture de EXP en binaire. - Le nombre peut être écrit : $M.10^0$ (M=MANTISSE) avec M< 1 (point décimal aligné à gauche) - Deux premiers digits du nombre contenus dans BCD1.	→ . 1 1 0 0 0 0 0 0 → 192 = 0 − 10^0=1 E = 0 M BCD1 → 0 0 1 1 0 1 1 1 3 7
- Deux digits suivants du nombre contenu dans BCD2	BCD2 → 0 1 0 1 0 0 0 0 5 0
- Deux dernier digits contenus dans BCD3.	BCD3 → 0 0 0 0 0 0 0 0 0 0 N! = $0,375000 \times 10^0$

Exemple de codage et décodage des nombres en virgule flottante

Codage

Opération	Résultat
DETERMINATION DU FORMAT DE STOCKAGE DE X! = + 4,25078	
- Représentation normalisée ($M*x10^E$).	→ + $0,425678 \times 10^1$
- Calcul de EXP (signe de la mantisse >0 → bit 7 = 0, signe de l'exposant >0 → bit 6 = 1, valeur de l'exposant = 1).	EXP 41H = 65D 0 1 0 0 0 0 0 1
- Calcul de BCD1 (4 dans ½ octet supérieur, 2 dans ½ octet inférieur).	→ BCD1 42H = 66D 0 1 0 0 0 0 1 0
- Calcul de BCD2 (50H)	→ BCD2 50H = 80D 0 1 0 1 0 0 0 0
- Calcul de BCD3 (78H)	→ BCD3 78H = 120D 0 1 1 1 1 0 0 0
DETERMINATION DU FORMAT DE STOCKAGE DE X! = - 0,125	
- Représentation normalisée ($M*x10^E$).	→ - $0,125000 \times 10^0$
- Calcul de EXP (signe de la mantisse <0 → bit 7 = 1, signe de l'exposant >0 → bit 6 = 1, valeur de l'exposant = 0).	EXP C0H = 192D 1 1 0 0 0 0 0 0
- Calcul de BCD1 (1 dans ½ octet supérieur, 2 dans ½ octet inférieur).	→ BCD1 12H = 18D 0 0 0 1 0 0 1 0
- Calcul de BCD2 (5 dans ½ octet supérieur, 0 dans ½ octet inférieur).	→ BCD2 50H = 80D 0 1 0 1 0 0 0 0
- Calcul de BCD3 (0 dans ½ octet supérieur, 0 dans ½ octet inférieur).	→ BCD3 00H = 0D 0 0 0 0 0 0 0 0

* *Avec M <1 (virgule décimale ou point décimal anglo-saxon aligné sur la gauche).*

POINTEUR =	F6C4H-F6C5H 63172D-63173D

Rôle

Stockage des variables ayant reçu un dimensionnement (implicite ou explicite). La table considérée mémorise le nom, le type, la valeur de tous les éléments du tableau.

Format - Variables numériques (E, SP, DP)

N° octet (DEC)	*Contenu*
1	- ITD de la variable considérée 02H 04H=SP 08H=DP
2	- Premier caractère du nom de la variable (code ASCII) (00H si n'existe pas).
3	- Second caractère du nom de la variable (code ASCII)
4	- OMS de la longueur du tableau (représentatif du nombre d'octets nécessaires pour parvenir à la variable tableau suivante).
5	- OPS de la longuer du tableau : longueur tableau = OMS + 256 * OPS (nombre d'octets du tableau après OCTET 5 et avant OCTET 1 de la variable suivante).
6	- Nombre de dimensions du tableau (ou nombre d'index) 0 < n < 255
7 à n	- (valeur maximale de chaque dimension +1) → 2 octets sont nécessaires pour chaque dimension par un stockage de la forme (OMS, OPS). Les valeurs max des dimensions sont données successivement pour les index de droite à gauche. *Remarque 1* : le nombre d'octets minimum : 2 pour tableau à une dimension.
n+1 à m	- Valeur de chaque élément du tableau en partant de la valeur minimum de l'index le plus à gauche pour arriver à la valeur maximale de l'index le plus à droite(*). - Chaque valeur d'élément nécessite 2, 3, 4, 8 octets** suivant l'ITD contenu dans l'octet 1 (nombre d'octets minimum = 2 pour tableau à 1 dimension et à 1 élément).

** Le format de stockage est identique à celui des variables simples (E = 2 octets, CC = 3 octets, SP = 4 octets, DP = 8 octets).

* Ceci signifie, par exemple, qu'avec un tableau du type A (2,2) l'ordre de mémorisation du bas vers le haut de la mémoire sera le suivant :

A(0,0) → A(1,0) → A(2,0) → A(0,1) → A(1,1) → A(2,1)
A(0,2) → A(1,2) → A(2,2)

Le nombre d'octets minimum, pour le stockage d'une variable tableau sera de :

$\underbrace{6}_{\text{octets 1 à 6}} + \underbrace{2}_{\text{octets 7 à n}} + \underbrace{2x2}_{\text{octets n+1 à m}} = 12$ octets+ {Exemple variable A% (1)}

\+ en partant du principe que le tableau entier le plus restreint qui ne puisse être confondu avec une variable simple comporte au moins 1 dimension et 2 éléments : A%(0) et A%(1).

ADRESSES RAM

Format variables chaînes de caractères (CC)

N° octet (DEC)	Contenu
1	- ITD variable chaîne de caractères = 03H
2	- Premier caractère du nom de la variable (code ASCII)
3	- Second caractère du nom de la variable (code ASCII).
4	- OMS de la longueur du tableau (après octet 5).
5	- OPS de la longueur du tableau L = OMS + 256 x OPS.
6	- Nombre de dimensions du tableau
7 à n	- (valeur maximale de chaque dimension +1) 2 octets sont nécessaires pour chaque dimension pour un stockage de la forme OMS, OPS. Les valeurs max de dimensions sont données successivement pour les index de droite à gauche. *Remarque :* le nombre d'octets minimum = 2 pour tableau à une dimension.
n+1 à m	1 Longueur de chaque élément CC du tableau 2 OMS de l'adresse du 1e caractère de chaque élément CC 3 OPS de l'adresse du 1e caractère de chaque élément CC (identique à variable simple type CC)
*	Ce processus est renouvelé autant de fois qu'il y a d'éléments constitutifs CC dans le tableau → A$(n,m) → nombre octets = (n+1) x (m+1) x 3.

* Le processus d'analyse pour une variable A$ (2,2) est :

A$(0,0) → A$(1,0) → A$(2,0) → A$(0,1) → A$(1,1)
A$(2,1) → A$(0,2) → A$(1,2) → A$(2,2)

Le nombre d'octets minimum, pour le stockage d'une variable tableau CC est :

$$\underbrace{6}_{\text{octets 1 à 6}} + \underbrace{2}_{\text{octets 7 à n}} + \underbrace{2 \times 3}_{\text{octets n+1 à m}} = 14$$ (exemple A$(1))

Formule de calcul du nombre d'octets nécessaire pour stocker un tableau (E, CC, SP, DP)

soit le tableau A (N_1, N_2... N_K)

$$N = 6 + (K \times 2) + \{(N_1+1) \times (N_2+1) \times \ldots (N_K+1)\} \times ITD$$

avec ITD = 2 (E) et K = nombre de dimensions dans le tableau
3 (CC)
4 (SP)
8 (DP)

Exemples

Variable	N° octet	Contenu HEX	Contenu DEC	Signification
A%(1,1)	1	02	2	Indicateur type entier (ITD = 02H)
	2	41	65	Premier caractère du nom de la variable 41H → A
avec	3	00	0	Second caractère du nom de la variable (non existant)
A%(0,0)=16	4	0D	13	OMS de l'offset (ou de la longueur du tableau après octet 5) = 13D
A%(1,0)=257	5	00	0	OPS de la longueur du tableau = 00 car L < 255
A%(0,1)=1024	6	02	2	Nombre de dimensions du tableau = 2
A%(1,1)=8193	7	02	2	OMS de la (valeur maximale +1) de l'index de droite
	8	00	0	OPS de la (valeur maximale +1) de l'index de droite
	9	02	2	OMS de la (valeur maximale +1) de l'index de gauche
	10	00	0	OPS de la (valeur maximale +1) de l'index de gauche
	11	10	16	OMS de la valeur A(0,0) du tableau
Nombre	12	00	0	OPS de la valeur A(0,0) du tableau = 00H car N < 256
octets	13	01	1	OMS de la valeur A(1,0) du tableau } 257 = 1 + 1 x 256
= 18	14	01	1	OPS de la valeur A(1,0) du tableau
	15	00	0	OMS de la valeur A(0,1) du tableau } 1024 = 0 + 4 x 256
	16	04	4	OPS de la valeur A(0,1) du tableau
	17	01	1	OMS de la valeur A(1,1) du tableau } 8193 = 1 + 32 x 256
	18	20	32	OPS de la valeur A(1,1) du tableau
A!(2)	1	04	4	ITD pour nombre SP
avec	2	00	0	Premier caractère du nom de la variable → 41H A
A!(0)=0,125	3	41	65	Second caractère du nom de la variable (non existant)
A!(1)=2,5	4	0F	15	OMS de la longueur du tableau (après octet 5)
A!(2)=-6,25	5	00	0	OPS de la longueur du tableau → L = OMS + 256 x OPS
	6	01	1	Nombre de dimensions du tableau = 1
	7	03	3	OMS de la (valeur maximale +1) de l'index
	8	00	0	OPS de la (valeur maximale +1) de l'index

(OFFSET = 13D : octets 6 à 18 de A%(1,1) ; OFFSET = 15D : à partir de l'octet 6 de A!(2))

Variable	N° octet	Contenu HEX	Contenu DEC	Signification
	9	40	64	EXP de la valeur A(0) = 0,125
	10	12	18	BCD1 de la valeur A(0) = 0,125 } Voir exemple de variables simples SP
	11	50	80	BCD2 de la valeur A(0) = 0,125
	12	00	0	BCD3 de la valeur A(0) = 0,125
Nombre	13	41	65	EXP de la valeur A(1) = 2,5
octets	14	25	37	BCD1 de la valeur A(1) = 2,5 } Voir exemples de variables simples SP
= 20	15	00	0	BCD2 de la valeur A(1) = 2,5
	16	00	0	BCD3 de la valeur A(1) = 2,5
	17	C1	193	EXP de la valeur A(2) = -6,25
	18	62	98	BCD1 de la valeur A(2) = -6,25 } Voir exemples de variables simples SP
	19	50	80	BCD2 de la valeur A(2) = -6,25
	20	00	0	BCD3 de la valeur A(2) = -6,25
Z$(3)	1	03	3	ITD pour chaîne de caractères
	2	5A	90	Premier caractère du nom de la variable 5AH → Z
Z$(0)="MSX64"	3	00	0	Second caractère du nom de la variable
	4	0F	15	OMS de la longueur du tableau (après octet 5)
Z$(1)="Modèle 2"	5	00	0	OPS de la longueur du tableau → L = OMS + 256 x OPS
	6	01	1	Nombre de dimensions du tableau 1
Z$(2)="BASIC"	7	04	4	OMS de la (valeur maximale +1) de l'index
	8	00	0	OPS de la (valeur maximale +1) de l'index
	9	05	5	Longueur de Z$(0) = 5
Z$(3)=	10	14	20	OMS de l'adresse du 1er caractère de Z$(0)
"VERS 1.4"	11	80	128	OPS de l'adresse du 1er caractère de Z$(0) → Ad = OMS + 256 x OPS, Ad = 8014H = 32788D

(OFFSET = 15D : suite du tableau A!(2), jusqu'à l'octet 20 ; OFFSET = 15D : à partir de l'octet 6 de Z$(3))

Variable	N° octet	Contenu HEX	Contenu DEC	Signification
Nombre octets = 20	12	07	7	Longueur de Z$(1) = 7
	13	22	34	OMS de l'adresse du 1er caractère de Z$(1)
	14	80	128	OPS de l'adresse du 1er caractère de Z$(1) - Ad = 8022H*
	15	05	5	Longueur de Z$(2) = 5
	16	32	50	OMS de l'adresse du 1er caractère de Z$(2)
	17	80	128	OPS de l'adresse du 1er caractère de Z$(2) - Ad = 8032H*
	18	07	7	Longueur de Z$(3) = 7
	19	40	64	OMS de l'adresse du 1er caractère de Z$(3)
	20	80	128	OPS de l'adresse du 1er caractère de Z$(3) - Ad = 8040H*

* 8014H = 32788D → PRINT PEEK (32788) → ("M") 1ère lettre de MSX 64
* 8022H = 32802D → PRINT PEEK (32802) → ("M") 1ère lettre de MODELE 2
* 8032H = 32818D → PRINT PEEK (32818) → ("B") 1ère lettre de BASIC
* 8040H = 32826D → PRINT PEEK (32826) → ("Y") 1ère lettre du VERS 1.4

POINTEUR =	F674H-F675H 63092D-63093D

Pointeur de la pile opérationnelle

Rôle

La "pile" est un registre de type LIFO (LAST IN - FIRST OUT → premier octet entré - premier octet sorti). Elle a pour but d'assurer un stockage temporaire d'informations lors de la rupture du déroulement normal d'un programme (assembleur Basic ...).

En langage d'assemblage Z80, tout appel à sous-programme (instruction CALL ou RST) provoque la "mise en pile" automatique (instruction manuelle PUSH), tout retour de sous-programme (instruction RET) provoque le "retrait de la pile" automatique (instruction nouvelle POP).

En Basic, certaines instructions, de par leur nature, utilisent la pile ; c'est le cas de FOR ... NEXT, du fait de sa structure nécessitant une mémorisation temporaire, c'est le cas également de GOSUB.

Sous Basic, la "pile" possède une "base" située à la fin de la "zone de chaîne de caractères" (ZCC) (haut de mémoire) et un sommet à allocation dynamique orienté vers le bas de la mémoire en tampon avec la "zone mémoire libre" (ZML).

PILE OPERATIONNELLE (PO)

Format 1 pour instructions : FOR...NEXT
Taille = 25 octets

↑ vers le bas de la mémoire
= sommet de la pile

N° octet	*Contenu*		
1	Token du mot-clé Basic "FOR" = 130D = 82H		
2 3	OMS de l'adresse de l'index FOR OPS de l'adresse de l'index FOR	(Ad) DEC = OMS + 256 x OPS	
4 + 5	Signe de l'incrément (PAS) de la boucle + 05H (1 si signe > 0, -1 = FFH si signe < 0)		
6 - 10 7 - 11 8 - 12 9 - 13	EXP de la valeur du PAS (STEP) BCD1 de la valeur du PAS (STEP) BCD2 de la valeur du PAS (STEP) BCD3 de la valeur du PAS (STEP)	BCD4 BCD5 BCD6 BCD7	Format DP (8 octets)
14 - 18 15 - 19 16 - 20 17 - 21	EXP de la valeur du nombre de boucles BCD1 de la valeur du nombre de boucles BCD2 de la valeur du nombre de boucles BCD3 de la valeur du nombre de boucles	BCD4 BCD5 BCD6 BCD7	Format DP 8 octets
22 23	OMS du n° de ligne où apparaît FOR OPS du n° de ligne où apparaît FOR	N° = OMS + 256 x OPS	
24 25	OMS de l'adresse de la 1ère instruction de boucle OPS de l'adresse de la 1ère instruction de boucle		

vers le haut de la mémoire
= base de la pile

Format 2 pour instruction GOSUB
Taille = 5 octets

↑ vers le bas de la mémoire
= sommet de la pile

N° octet	*Contenu*
1	Token du mot-clé "GOSUB" = 141D = 8DH
2 3	OMS du n° de la ligne où apparaît le token "GOSUB" OPS du n° de la ligne où apparaît le token "GOSUB"
4 5	OMS de l'adresse de la ligne avec GOSUB dans TIP OPS de l'adresse de la ligne avec GOSUB dans TIP

vers le haut de la mémoire
= base de la pile

ANNEXE

ANNEXE

TABLE DE CONVERSION DE BASES

Dec	Binaire	Hex	Octal	Dec	Binaire	Hex	Octal
Ø	ØØØØØØØØ	ØØ	ØØØ	39	ØØ1ØØ111	27	Ø47
1	ØØØØØØØ1	Ø1	ØØ1	4Ø	ØØ1Ø1ØØØ	28	Ø5Ø
2	ØØØØØØ1Ø	Ø2	ØØ2	41	ØØ1Ø1ØØ1	29	Ø51
3	ØØØØØØ11	Ø3	ØØ3	42	ØØ1Ø1Ø1Ø	2A	Ø52
4	ØØØØØ1ØØ	Ø4	ØØ4	43	ØØ1Ø1Ø11	2B	Ø53
5	ØØØØØ1Ø1	Ø5	ØØ5	44	ØØ1Ø11ØØ	2C	Ø54
6	ØØØØØ11Ø	Ø6	ØØ6	45	ØØ1Ø11Ø1	2D	Ø55
7	ØØØØØ111	Ø7	ØØ7	46	ØØ1Ø111Ø	2E	Ø56
8	ØØØØ1ØØØ	Ø8	Ø1Ø	47	ØØ1Ø1111	2E	Ø57
9	ØØØØ1ØØ1	Ø9	Ø11	48	ØØ11ØØØØ	3Ø	Ø6Ø
1Ø	ØØØØ1Ø1Ø	ØA	Ø12	49	ØØ11ØØØ1	31	Ø61
11	ØØØØ1Ø11	ØB	Ø13	5Ø	ØØ11ØØ1Ø	32	Ø62
12	ØØØØ11ØØ	ØC	Ø14	51	ØØ11ØØ11	33	Ø63
13	ØØØØ11Ø1	ØD	Ø15	52	ØØ11Ø1ØØ	34	Ø64
14	ØØØØ111Ø	ØE	Ø16	53	ØØ11Ø1Ø1	35	Ø65
15	ØØØØ1111	ØF	Ø17	54	ØØ11Ø11Ø	36	Ø66
16	ØØØ1ØØØØ	1Ø	Ø2Ø	55	ØØ11Ø111	37	Ø67
17	ØØØ1ØØØ1	11	Ø21	56	ØØ111ØØØ	38	Ø7Ø
18	ØØØ1ØØ1Ø	12	Ø22	57	ØØ111ØØ1	39	Ø71
19	ØØØ1ØØ11	13	Ø23	58	ØØ111Ø1Ø	3A	Ø72
2Ø	ØØØ1Ø1ØØ	14	Ø24	59	ØØ111Ø11	3B	Ø73
21	ØØØ1Ø1Ø1	15	Ø25	6Ø	ØØ1111ØØ	3C	Ø74
22	ØØØ1Ø11Ø	16	Ø26	61	ØØ1111Ø1	3D	Ø75
23	ØØØ1Ø111	17	Ø27	62	ØØ11111Ø	3E	Ø76
24	ØØØ11ØØØ	18	Ø3Ø	63	ØØ111111	3F	Ø77
25	ØØØ11ØØ1	19	Ø31	64	Ø1ØØØØØØ	4Ø	1ØØ
26	ØØØ11Ø1Ø	1A	Ø32	65	Ø1ØØØØØ1	41	1Ø1
27	ØØØ11Ø11	1B	Ø33	66	Ø1ØØØØ1Ø	42	1Ø2
28	ØØØ111ØØ	1C	Ø34	67	Ø1ØØØØ11	43	1Ø3
29	ØØØ111Ø1	1D	Ø35	68	Ø1ØØØ1ØØ	44	1Ø4
3Ø	ØØØ1111Ø	1E	Ø36	69	Ø1ØØØ1Ø1	45	1Ø5
31	ØØØ11111	1F	Ø37	7Ø	Ø1ØØØ11Ø	46	1Ø6
32	ØØ1ØØØØØ	2Ø	Ø4Ø	71	Ø1ØØØ111	47	1Ø7
33	ØØ1ØØØØ1	21	Ø41	72	Ø1ØØ1ØØØ	48	11Ø
34	ØØ1ØØØ1Ø	22	Ø42	73	Ø1ØØ1ØØ1	49	111
35	ØØ1ØØØ11	23	Ø43	74	Ø1ØØ1Ø1Ø	4A	112
36	ØØ1ØØ1ØØ	24	Ø44	75	Ø1ØØ1Ø11	4B	113
37	ØØ1ØØ1Ø1	25	Ø45	76	Ø1ØØ11ØØ	4C	114
38	ØØ1ØØ11Ø	26	Ø46	77	Ø1ØØ11Ø1	4D	115

Dec	Binaire	Hex	Octal	Dec	Binaire	Hex	Dec
78	01001110	4E	116	125	01111101	7D	175
79	01001111	4F	117	126	01111110	7E	176
80	01010000	50	120	127	01111111	7F	177
81	01010001	51	121	128	10000000	80	200
82	01010010	52	122	129	10000001	81	201
83	01010011	53	123	130	10000010	82	202
84	01010100	54	124	131	10000011	83	203
85	01010101	55	125	132	10000100	84	204
86	01010110	56	126	133	10000101	85	205
87	01010111	57	127	134	10000110	86	206
88	01011000	58	130	135	10000111	87	207
89	01011001	59	131	136	10001000	88	210
90	01011010	5A	132	137	10001001	89	211
91	01011011	5B	133	138	10001010	8A	212
92	01011100	5C	134	139	10001011	8B	213
93	01011101	5D	135	140	10001100	8C	214
94	01011110	5E	136	141	10001101	8D	215
95	01011111	5F	137	142	10001110	8E	216
96	01100000	60	140	143	10001111	8F	217
97	01100001	61	141	144	10010000	90	220
98	01100010	62	142	145	10010001	91	221
99	01100011	63	143	146	10010010	92	222
100	01100100	64	144	147	10010011	93	223
101	01100101	65	145	148	10010100	94	224
102	01100110	66	146	149	10010101	95	225
103	01100111	67	147	150	10010110	96	226
104	01101000	68	150	151	10010111	97	227
105	01101001	69	151	152	10011000	98	230
106	01101010	6A	152	153	10011001	99	231
107	01101011	6B	153	153	10011010	9A	232
108	01101100	6C	154	155	10011011	9B	233
109	01101101	6D	155	156	10011100	9C	234
110	01101110	6E	156	157	10011101	9D	235
111	01101111	6F	157	158	10011110	9E	236
112	01110000	70	160	159	10011111	9F	237
113	01110001	71	161	160	10100000	A0	240
114	01110010	72	162	161	10100001	A1	241
115	01110011	73	163	162	10100010	A2	242
116	01110100	74	164	163	10100011	A3	243
117	01110101	75	165	164	10100100	A4	244
118	01110110	76	166	165	10100101	A5	245
119	01110111	77	167	166	10100110	A6	246
120	01111000	78	170	167	10100111	A7	247
121	01111001	79	171	168	10101000	A8	250
122	01111010	7A	172	169	10101001	A9	251
123	01111011	7B	173	170	10101010	AA	252
124	01111100	7C	174	171	10101011	AB	253

Dec	Binaire	Hexa	Octal	Dec	Binaire	Hexa	Octal
172	10101100	AC	254	214	11010110	D6	326
173	10101101	AD	255	215	11010111	D7	327
174	10101110	AE	256	216	11011000	D8	330
175	10101111	AF	257	217	11011001	D9	331
176	10110000	B0	260	218	11011010	DA	332
177	10110001	B1	261	219	11011011	DB	333
178	10110010	B2	262	220	11011100	DC	334
179	10110011	B3	263	221	11011101	DD	335
180	10110100	B4	264	222	11011110	DE	336
181	10110101	B5	265	223	11011111	DF	337
182	10110110	B6	266	224	11100000	E0	340
183	10110111	B7	267	225	11100001	E1	341
184	10111000	B8	270	226	11100010	E2	342
185	10111001	B9	271	227	11100011	E3	343
186	10111010	BA	272	228	11100100	E4	344
187	10111011	BB	273	229	11100101	E5	345
188	10111100	BC	274	230	11100110	E6	346
189	10111101	BD	275	231	11100111	E7	347
190	10111110	BE	276	232	11101000	E8	350
191	10111111	BF	277	233	11101001	E9	351
192	11000000	C0	280	234	11101010	EA	352
193	11000001	C1	281	235	11101011	EB	353
194	11000010	C2	282	236	11101100	EC	354
195	11000011	C3	283	237	11101101	ED	355
196	11000100	C4	284	238	11101110	EE	356
197	11000101	C5	285	239	11101111	EF	357
198	11000110	C6	286	240	11110000	F0	360
199	11000111	C7	287	241	11110001	F1	361
200	11001000	C8	290	242	11110010	F2	362
201	11001001	C9	291	243	11110011	F3	363
202	11001010	CA	292	244	11110100	F4	364
203	11001011	BB	293	245	11110101	F5	365
204	11001100	CC	294	246	11110110	F6	366
205	11001101	CD	295	247	11110111	F7	367
206	11001110	CE	296	248	11111000	F8	370
207	11001111	CF	297	249	11111001	F9	371
208	11010000	D0	300	250	11111010	FA	372
209	11010001	D1	301	251	11111011	FB	373
210	11010010	D2	302	252	11111100	FC	374
211	11010011	D3	303	253	11111101	FD	375
212	11010100	D4	304	254	11111110	FE	376
213	11010101	D5	305	255	11111111	FF	377

Français - Anglais

Sigle français	Signification	Sigle anglais	Signification
A	Adresse	A	Address
ACC	Accumulateur	ACC	Accumulator
A/N	Analogique/Numérique	A/D	Analog to Digital
ASCII	Code Standard américain pour l'échange d'informations	ASCII	American Standard Code for Information Interchange
BASIC	Code d'instruction à usage multiple destiné aux débutants	BASIC	Beginners All purpose Symbolic Instruction Code
BCF	Bloc de Contrôle de Fichier	FCB	File Control Block
BIOS	Entrée/Sortie Système de base	BIOS	Basic Input Output System
BIT	Elément binaire	BIT	Binary Digit
BIT/S	Bits par Seconde	BPS	Bit Per Second
BMS	Bit le Moins Significatif	LSB LOB	Least Significant Bit Lowest Order Bit
BPS	Bit le Plus Significatif	MSB HOB	Most Significant Bit Highest Order Bit
C	Retenue (CIE-Z80)	C	Carry (CCR-Z80)
CAR/L	Caractères par ligne	CPL	Character Per Line
CAR/S	Caractères par Seconde	CPS	Character Per Second
CC	Chaîne de caractères	$	String
CO	Compteur Ordinal	PC	Program Counter
D	Donnée	D	Data
DCB	Décimal Codé Binaire	BCD	Binary Coded Decimal
DEC	Décimal	DEC	Decimal
DI	Interdiction des Interruptions	DI	Disable Interrupt
DP	Double précision	DP	Double precision
E	Entier	I	Integer

Français - Anglais

Sigle français	Signification	Sigle anglais	Signification
EI	Validation des interruptions	EI	Enable Interrupt
FE	Fin d'Enregistrement	EOR	End Of Record
FF	Fin de Fichier	EOF	End Of File
FI	Fin d'Instruction	EOS	End Of Statement
FIFO	Premier entré - Premier sorti	FIFO	First In - First Out (register)
FL	Fin de Ligne	EOL	End Of Line
H	Demi retenue (RIE-Z80)	H	Half carry (CCR-Z80)
HEX	Hexadécimal	HEX	Hexadecimal
HTR	Horloge en Temps Réel	RTC	Real Time Clock
I	Imprimante	LPT	Line Printer
INT	Interruption	INT	Interrupt
IRQ	Demande d'Interruption	IRQ	Interrupt Request
ITD	Indicateur de Type de Données	DTF NTF	Data Type Flag Number Type Flag
LIFO	Dernier entré - premier sorti	LIFO	Last In - First Out
LF	Interligne	LF	Line Feed
L/E	Lecture - Ecriture	R/W	Read - Write
MpU	Unité microprocesseur	MpU	Microprocessing Unit
MSX	Microsoft Etendu	MSX	Microsoft extended
N	Negatif (RIE-Z80)	N	Negative (CCR-Z80)
N/A	Numérique - Analogique	D/A	Digital to Analog
NMI	Interruption Non Masquable (Z80)	NMI	Non Maskable Interrupt (Z80)
OMS	Octet le Moins Significatif	LSB	Least Significant Byte
OPS	Octet le Plus Significatif	MSB	Most Significant Byte
P	Parité (RIE-Z80)	P	Parity (CCR-Z80)

Français-Anglais

Sigle français	*Signification*	*Sigle anglais*	*Signification*
PP	Pointeur de Pile	**SP**	Stack Pointer
RAM	Mémoire à lecture/ écriture	**RAM**	Random Access Memory
RC	Retour Chariot	**CR**	Carriage Return
RIE	Registre des indicateurs d'Etat	**CCR**	Code Condition Register
ROM	Mémoire à lecture seule	**ROM**	Read Only Memory
RST	Redémarrage (Z80)	**RST**	Restart (Z80)
SED	Système d'Exploitation Disque	**DOS**	Disk Operating System
SP	Simple Précision	**SP**	Simple Precision
SSP	Synthétiseur sonore Programmable	**PSG**	Programmable Sound Generator
TAS	Table des attributs des Sprites	**SAT**	Sprite Attribute Table
TC	Table des couleurs	**CT**	Color Table
TGC	Table du Générateur de Configuration	**PGT**	Pattern Generator Table
TGS	Table du Générateur des Sprites	**SGT**	Sprite Generator Table
TIP	Table des Instructions de Programme	**PST**	Program Statement Table
TNC	Table des Noms de Configuration	**PNT**	Pattern Name Table
TVS	Table des variables Simples	**VLT**	Variable List Table
TVT	Table des variables	**VLT**	Variable List Table
V	Dépassement de capacité (RIE - Z80)	**V**	Overflow (CCR - Z80)
XOR	OU Exclusif	**XOR**	Exclusive OR
Z	Zéro (RIE - Z80)	**Z**	Zero (CCR - Z80)
ZML	Zone Mémoire Libre	**FSL**	Free Space List

Anglais-Français

Sigle anglais	*Signification*	*Sigle français*	*Signification*
A	Address.	**A**	Adresse
ACC	Accumulator	**ACC**	Accumulateur
A/D	Analog to Digital	**A/N**	Analogique-Numérique
ASCII	American Standard Code for Information Interchange	**ASCII**	Code Standard Américain pour l'échange d'Informations
BASIC	Beginners' All purpose Symbolic Instruction Code	**BASIC**	Code d'Instruction à usage multiple destiné aux débutants
BCD	Binary Coded Decimal	**DCB**	Décimal Codé en Binaire
BIOS	Basic Input Output System	**BIOS**	E/S Système de Base
BIT	Binary Digit	**BIT**	Bit (élément binaire)
BPS	Bits Per Second	**BIT/S**	Bits par Seconde
C	Carry (CCR Z80)	**C**	Retenue (RIE - Z80)
CCR	Code Condition Register	**RIE**	Registre des Indicateurs d'Etat
CPL	Character Per Line	**CAR/L**	Caractères par Ligne
CPS	Character Per Second	**CAR/S**	Caractères par Seconde
CPU	Central Processing Unit	**UCT**	Unité Centrale de Traitement
CR	Carriage Return	**RC**	Retour Chariot
CRT	Cathode Ray Tube	**TC**	Tube Cathodique (moniteur vidéo)
CT	Color Table	**TC**	Table des Couleurs
D	Data	**D**	Donnée
D/A	Digital to Analog	**N/A**	Numérique - Analogique
DEC	Decimal	**DEC**	Décimal
DI	Disable Interrupt (Z80)	**DI**	Interdiction d'Interruption (Z80)
DOS	Operating System	**SED**	Système d'Exploitation Disque

Anglais-Français

Sigle Anglais	*Signification*	*Sigle français*	*Signification*
DP	Double Precision	DP	Double Précision
DTF	Data Type Flag	ITD	Indicateur de Type de Données
EI	Enable Interrupt	EI	Validation ou autorisation d'interruption
EOF	End Of File	FF	Fin de Fichier
EOL	End OF Line	FL	Fin de ligne
EOR	End Of Record	FE	Fin d'Enregistrement
EOS	End Of Statement	FI	Fin d'Instruction
EOT	End Of Text	FT	Fin de Texte
FCB	File Control Block	BCF	Bloc de Contrôle de Fichier
FIFO	First In - First Out	FIFO	Premier entré - premier sorti (registre)
FSL	Free Space List	ZML	Zone Mémoire Libre
H	Half Carry (CCR-Z80)	H	Demi-retenue (RIE-Z80)
HEX	Hexadecimal	HEX	Hexadécimal
HOB	Highest Order Bit	BPS	Bit le Plus Significatif
I	Integer	E	Entier
INT	Interrupt	INT	Interruption
I/O	Input - Output	E/S	Entrée - Sortie
IRQ	Interrupt Request	IRQ	Demande d'Interruption
KBD	Keyboard	KBD	Entrée Clavier
LIFO	Last In - First Out (register)	LIFO	Dernier entré - premier sorti (registre)
LF	Line Feed	LF	Interligne
LOB	Lowest Order Bit	BMS	Bit le Moins Significatif
LPT	Line Printer	I	Imprimante

Anglais-Français

Sigle anglais	*Signification*	*Sigle français*	*Signification*
LSB	Least Significant Bit (1)	BMS	Bit le Moins Significatif
	Byte (2)	OMS	Octet le Moins Significatif
MPU	Microprocessing Unit	MPU	Unité Microprocesseur
MSB	Most Significant Bit (1)	BPS	Bit le Plus Significatif
	Byte (2)	OPS	Octet le Plus Significatif
MSX	Microsoft Extended	MSX	Microsoft étendu
N	Negative (CCR-Z80)	N	Négatif (RIE-Z80)
NMI	Non Maskable Interrupt (Z80)	NMI	Interruption non Masquable
NTF	Number Type Flag	ITD	Indicateur de Type de Donnée
P	Parity (CCR-Z80)	P	Parité (RIE-Z80)
PC	Program Counter	CO	Compteur Ordinal
PGT	Pattern Generator Table	TGC	Table du Générateur de Configuration
PNT	Pattern Name Table	TNC	Table des Noms de Configuration
PST	Program Statement Table	TIP	Table des Instructions de Programme
PSW	Program Status Word (=CCR)	RIE	Registre des Indicateurs d'Etat
R	Read	L	Lecture
RAM	Random Access Memory	RAM	Mémoire à écriture/lecture
ROM	Read Only Memory	ROM	Mémoire à lecture seule
RST	Restart (Z80)	RST	Redémarrage (Z80)
R/W	Read/Write	L/E	Lecture/Ecriture
SAT	Sprite Attribute Table	TAS	Table des Attributs des Sprites

Anglais-Français

Sigle anglais	Signification	Sigle français	Signification
SGT	Sprite Generator Table	**TGS**	Table du Générateur des Sprites
SP	Simple Precision (1)	**SP**	Simple Précision
	Stack Pointer (2)	**PP**	Pointeur de Pile
V	Overflow (CCR-Z80)	**V**	Dépassement (RIE-Z80)
VLT	Variable List Table	**TV**	Table des Variables
		TVS	Table des Variables Simples
		TVT	Table des Variables Tableaux
XOR	Exclusive OR	**XOR**	OU Exclusif
Z	Zero (CCR-Z80)	**Z**	Zéro (RIE-Z80)

Mot	Signification
ASCII	Procédé de codage normalisé permettant d'exprimer les caractères alphanumériques d'un texte sous la forme de codes hexadécimaux, les fonctions (avance curseur, retour curseur etc.) sont également traduites en code ASCII.
BCF	Bloc de données en mémoire vive faisant la jonction d'E/S entre un programme et un fichier cassette. Ce lien est créé par la fonction OPEN et interrompu par la fonction CLOSE. Ce bloc est utilisé par toutes les fonctions d'E/S sous Basic, il contient les informations suivantes : champs EOF et NEXT, l'adresse du buffer d'E/S, longueur d'enregistrement, etc.
BUFFER	(ou TAMPON d'E/S). Espace en mémoire vive utilisé pour stocker le contenu d'un secteur lu sur une cassette ou d'un enregistrement à écrire sur une cassette. Chaque BCF ouvert (voir ci-dessus) utilise un buffer. Ce buffer constitue donc une zone de transit entre la mémoire centrale ROM-RAM résidente du système et la mémoire de masse magnétique externe.
CHAINE DE CARACTERES (abréviation CC)	Suite linéaire de caractères dont le sens doit être interprété textuellement. Des opérations peuvent être exécutées sur les chaînes. Fractionnement (MID$, LEFT$, RIGTH$) addition (concaténation), tests logiques etc.
FE	La "fin d'enregistrement" dans un fichier peut être décelée par une marque (configuration spéciale d'octets) placée juste après le dernier octet de l'enregistrement (c'est le cas pour les programmes Basic stockés sous format condensé par la marque 00H et la marque 0DH pour les fichiers Basic stockés sous format ASCII).
FF	La "fin de fichier" peut être détectée par une marque spécifique située en fin de fichier (ex. 00 00 H pour les fichiers Basic tokénisés).
FI	La "fin d'instruction" sous Basic est indiquée par l'utilisateur avec le symbole ":" dans le cas de lignes Basic à plusieurs instructions.
FIFO	Voir LIFO.

Mot	Signification
INTERRUP-TION	Signal provoquant la rupture de l'opération que l'ordinateur est en train d'effectuer afin d'obtenir l'exécution d'une autre tâche spécifique. L'ordinateur reprend généralement l'opération au point où elle s'était arrêtée après s'être acquitté de la tâche en question.
ITD	Code associé à un type de donnée spécifique (entier, simple précision, double précision, chaîne de caractères) utilisé par le Basic contenu en ROM au niveau des opérations sur des nombres stockés en mémoire RAM, les 4 codes possibles sont 02H==▷E 03H==▷CC 04H→SP 08H→DP.
LIFO	Structure type des "piles opérationnelles" où la dernière information entrée est la première sortie à opposer à la structure LIFO des "registres à décalage" où la première information entrée est la première sortie.
POINTEUR	Technique d'adressage indirect dans laquelle une adresse ne contient pas des octets de données mais des octets représentatifs d'une autre adresse. Un pointeur est généralement constitué de 2 octets, le premier vise la partie basse de l'adresse "pointée" (OMS) le second la partie haute (OPS) avec (adresse pointée) DEC = OMS + 256 x OPS. **La zone de communication** (F380H-FFFFH) contient de nombreux pointeurs (début TIP, début TVS, début TVT etc. --▷ Voir définition de ces mots).
PPI	**Programmable Port Interface.** Circuit intégré LSI µPD8255 utilisé sous MSX pour gérer de façon programmable différentes interfaces de périphériques (gestion des SLOTS, clavier, cassette).
PSG	**Programmable Sound Generator.** Circuit intégré LSI AY-3-8910 réalisant une synthèse sonore (convertisseur numérique/analogique) sur 3 voies d'une étendue de 8 octaves (+1 canal de bruit). L'ensemble des registres de ce circuit permettant une programmation totale des caractéristiques des signaux générés, il est également utilisé pour la gestion des manettes de jeu (Joysticks).

Mot	Signification
RAM	Equivalent : "mémoire vive" ou "mémoire à lecture-écriture". La mémoire vive est la zone où l'usager peut loger des données ou des programmes. Il existe deux types de mémoire RAM. Les mémoires statiques et les mémoires dynamiques qui nécessitent un circuit de rafraîchissement mais possèdent par leur structure un meilleur degré d'intégration. Elles sont utilisées dans le MSX dans la zone mémoire 8000H-FFFFH.
ROM	Equivalent : "mémoire morte" ou "mémoire à lecture seule". Elle se présente sous la forme de circuits intégrés à technologie MOS pré-programmés par le constructeur. Dans le cas du MSX l'interpréteur Basic est incorporé à une mémoire morte de 32 kilo-octets.
SLOT	(Signification littérale : **fente**) également appelé en français **"BANC MEMOIRE"** ou **"BANQUE MEMOIRE"** permet d'étendre la zone adressable du microprocesseur Z80A limité à 64K à nx64K par commutation spécifique. Quatre types de "SLOT" peuvent être distingués - SLOT 0 → ou BANC SYSTEME (ROM + RAM) - SLOT 1 → ou extension cartouche ROM - SLOT 2 - SLOT 3 Chaque SLOT peut lui-même recevoir quatre extensions. La gestion des "SLOTS" ou "BANCS MEMOIRE" est assurée par le circuit PPI.
SPRITE	(Littéralement **LUTIN** en français). Forme élémentaire s'inscrivant dans une matrice 8x8, 16x16 pouvant être animée à l'écran sous Basic sous 32 plans superposés avec priorité pour les lutins d'avant plan devant ceux d'arrière plan. Pour la création de jeux, la présence de lutins permet une grande facilité de programmation pour l'animation (ainsi que la détection de collisions).
TAS	**Table des Attributs des Sprites.** Cette table contient quatre paramètres pour chaque lutin. Le premier est la position verticale du lutin sur 1 octet, le second est la position hori-

Mot	Signification
	zontale sur un octet également le troisième paramètre est un pointeur vers la TGS définissant le dessin du LUTIN. Le dernier paramètre définit la couleur du LUTIN.
TC	**Table des Couleurs.** Une des cinq tables du processeur graphique VDP TMS 9929A ; elle est utilisée dans les modes graphiques 1 et 2 (SCREEN 1 et SCREEN 2). En mode graphique 1 la TC définit la couleur de chaque groupe de huit configurations (ou formes). Un octet est réservé par groupe. En mode graphique 2 la TC permet de définir deux couleurs pour chaque octet de la TGC, c'est-à-dire deux couleurs pour chaque groupe de huit points horizontaux.
TGC	**Table du Générateur de Configuration.** Cette table contient le dessin des 256 caractères ou configurations standards affichables. Elle peut être chargée avec des caractères alphanumériques ou semi-graphiques. Ces configurations peuvent être entièrement redéfinies par programme.
TGS	**Table du Générateur des Sprites.** Cette table contient la configuration du LUTIN, celui-ci étant défini sur 8x8 ou 16x16 bits (valeur définie dans registre 1).
TIP	**Table des Instructions de Programme.** C'est une zone de mémoire vive, située après la zone ROM, permettant de stocker les instructions d'un programme Basic sous forme condensé (tokenisé).
TNC	**Table des Noms de Configuration** En mode texte cette table contient le numéro de chaque caractère à afficher dans un ensemble de 960 mémoires contigües (40x24). La première mémoire contient la valeur du caractère à afficher en haut et à gauche de l'écran, le 40e en haut et à droite (fin 11e ligne). La dernière mémoire 960 : la valeur à afficher en bas à droite de l'écran. La TNC est également utilisée en mode graphique 1, graphique 2 et multicolore.

Mot	Signification
TVS	La **Table des Variables Simples** est une zone de mémoire vive située à la fin de la zone TIP permettant de stocker les noms et valeurs des variables non dimensionnées spécifiées dans la table des instructions de programme.
TVT	La **Table des Variables Tableaux** (ou Table des variables dimensionnées) est une zone de mémoire vive située à la fin de la zone TVS, permettant de stocker les noms et valeurs des variables dimensionnées spécifiées dans la TIP.
VDP	**Video Display Processor.** Circuit LSI jouant le rôle d'un processeur graphique spécialisé permettant l'affichage à l'écran couleur de la TV ou du moniteur en quatre modes distincts de 40x24 (mode texte) de définition jusqu'à 256x192 (mode graphique HR) permet également la définition des LUTINS ou SPRITES (voir ce mot).
VECTEUR	Technique utilisée en informatique pour accéder à une routine de traitement ou d'exécution par l'intermédiaire d'un Saut préalable (JUMP) à cette adresse. Le BIOS du MSX contient de nombreux **Vecteurs**, cette technique permet de maintenir dans une zone mémoire immuable un accès à des routines qui peuvent elles-mêmes varier.
VIDEORAM	ou **Mémoire Video** (en abrégé VRAM) Zone mémoire de 16KO indépendante de la mémoire utilisateur où sont stockées les images vidéo créées par VDP. Elle est constituée de 5 tables : TC, TAS, TGS, TGC, TNC (voir définitions de ces tables).
ZML	La **Zone Mémoire Libre** : est une zone de mémoire vive s'étendant de la fin de la zone TVT au sommet de la pile opérationnelle Basic, cette zone est utilisable par d'autres programmes (routines USR etc.).

CONSEILS DE LECTURE

Pour approfondir vos connaissances en BASIC MSX, et mieux connaître le système MSX, P.S.I. vous propose une palette d'ouvrages utiles.

Pour maîtriser le BASIC MSX

- BASIC MSX Méthodes Pratiques par Jacques Boisgontier (Editions du P.S.I.)

Pour ceux qui connaissent déjà bien le BASIC et souhaitent découvrir toutes les spécificités du MSX.

Pour mieux comprendre le système MSX

- Le livre du MSX par Daniel Martin (B.C.M.)

Etude poussée du fonctionnement interne du MSX, avec exemples en langage machine.

- Assembleur et périphériques des MSX par Pierre BRANDEIS et Frédéric BLANC (Editions du P.S.I.)

Pour vous initier aux principes de base de l'assembleur du Z80 et vous familiariser avec la VDP, le générateur de sons, le BIOS, la région de communication, etc.

- Programmer en Assembleur par Alain Pinaud (Editions du P.S.I.)

Introduction complète à la programmation en assembleur Z80, agrémentée de nombreux exercices et exemples.

Achevé d'imprimer en juin 1985
sur les presses de l'imprimerie Laballery et Cie — 58500 Clamecy
Dépôt légal : juin 1985

N° d'impression : 504087 N° d'édition : 86595-243-1 ISBN : 2-86595-243-6

Votre avis nous intéresse

- Pour nous permettre de faire de meilleurs livres, adressez-nous vos critiques sur le présent livre.
- Si vous souhaitez des éclaircissements techniques, écrivez-nous, nous adresserons votre demande à l'auteur qui ne manquera pas de vous répondre directement.

- Ce livre vous donne-t-il toute satisfaction?

- Y a-t-il un aspect du problème que vous auriez aimé voir abordé?

Comment avez-vous eu connaissance de ce livre?

☐ publicité ☐ cadeau
☐ catalogue ☐ librairie
☐ boutique micro ☐ exposition
☐ autres

Avez-vous déjà acquis des livres PSI?

lesquels? ______

qu'en pensez-vous? ______

Nom ______ Prénom ______ Age ______
Adresse ______
Profession ______
Centre d'intérêt ______

CATALOGUE GRATUIT

Vous pouvez obtenir un catalogue complet des ouvrages PSI, sur simple demande, ou en retournant cette page remplie à votre libraire, à votre boutique micro ou aux

Editions du PSI
BP 86
77402 Lagny-sur-Marne Cedex